

**PACLIC 2015**

**29th Pacific Asia Conference on Language,  
Information and Computation  
Proceedings of PACLIC 2015:  
Poster Papers**

**Program chair:**

**Hai Zhao**

**30 October - 1 November, 2015**

**Shanghai, China**

## **Sponsors**

Department of Computer Science and Engineering, Shanghai Jiao Tong University

Chinese Information Processing Society of China (CIPS)

LY Education Technology

Shanghai Computer Federation Artificial Intelligence Committee (SCFAIC)

## Table of Contents

Identification of Sympathy in Free Conversation <i>Tomotaka Fukuoka and Kiyooki Shirai</i> .....	1
Toward a Corpus of Cantonese Verbal Comments and their Classification by Multi-dimensional Analysis <i>Oi Yee Kwong</i> .....	10
Improved Entity Linking with User History and News Articles <i>Soyun Jeong, Youngmin Park, Sangwoo Kang and Jungyun Seo</i> .....	19
An Arguing Lexicon for Stance Classification on Short Text Comments in Chinese <i>Ju-Han Chuang and Shu-Kai Hsieh</i> .....	27
Learning Sentential Patterns of Various Rhetoric Moves for Assisted Academic Writing <i>Jim Chang, Hsiang-Ling Hsu, Joanne Boisson, Hao-Chun Peng, Yu-Hsuan Wu and Jason S. Chang</i> .....	37
Idioms: Formally Flexible but Semantically Non-transparent <i>Hee-Rahk Chae</i> .....	46
Asymmetries in Scrambling and Distinctness of Copies <i>Gwangrak Son</i> .....	55
Detecting an Infant's Developmental Reactions in Reviews on Picture Books <i>Hiroshi Uehara, Mizuho Baba and Takehito Utsuro</i> .....	64
Semi-automatic Filtering of Translation Errors in Triangle Corpus <i>Sung-Kwon Choi, Jong-Hun Shin and Young-Gil Kim</i> .....	72
Cross-language Projection of Dependency Trees for Tree-to-tree Machine Translation <i>Yu Shen, Chenhui Chu, Fabien Cromieres and Sadao Kurohashi</i> .....	80
Realignment from Finer-grained Alignment to Coarser-grained Alignment to Enhance Mongolian-Chinese SMT <i>Jing Wu, Hongxu Hou and Xie Congjiao</i> .....	89
Finding the Origin of a Translated Historical Document <i>Zahrul Islam and Natia Dundua</i> .....	96

Improving the Performance of an Example-Based Machine Translation System Using a Domain-specific Bilingual Lexicon <i>Nasredine Semmar, Othman Zennaki and Meriama Laib</i> .....	106
A Multifactorial Analysis of English Particle Movement in Korean EFL Learners' Writings <i>Gyu-Hyeong Lee, Ha-Eung Kim and Yong-Hun Lee</i> .....	116
An Efficient Annotation for Phrasal Verbs using Dependency Information <i>Masayuki Komai, Hiroyuki Shindo and Yuji Matsumoto</i> .....	125
Color Aesthetics and Social Networks in Complete Tang Poems: Explorations and Discoveries <i>Chao-Lin Liu, Hongsu Wang, Wen-Huei Cheng, Chu-Ting Hsu and Wei-Yun Chiu</i> ..	132
Korean Twitter Emotion Classification Using Automatically Built Emotion Lexicons and Fine-Grained Features <i>Hyo Jin Do and Ho-Jin Choi</i> .....	142
Chinese Word Segmentation based on analogy and majority voting <i>Zongrong Zheng, Yi Wang and Yves Lepage</i> .....	151
Enhancing Root Extractors Using Light Stemmers <i>Mahmoud El-Defrawy, Yasser El-Sonbaty and Nahla Belal</i> .....	157
Where Morphological Complexity Matters <i>Tomokazu Takehisa</i> .....	167
Distinguishing between True and False Stories using various Linguistic Features <i>Yaakov Hacoheh-Kerner, Rakefet Dilmon, Shimon Friedlich and Daniel Nisim Cohen</i> .....	176
Bilingually motivated segmentation and generation of word translations using relatively small translation data sets <i>Kavitha Karimbi Mahesh, Luis Gomes and José Lopes</i> .....	187
Selecting Contextual Peripheral Information for Answer Presentation: The Need for Pragmatic Models <i>Rivindu Perera and Parma Nand</i> .....	197
RealText-asg: A Model to Present Answers Utilizing the Linguistic Structure of Source Question <i>Rivindu Perera and Parma Nand</i> .....	206

Learning under Covariate Shift for Domain Adaptation for Word Sense Disambiguation	
<i>Hiroyuki Shinnou, Minoru Sasaki and Kanako Komiya</i> .....	215
Unsupervised Domain Adaptation for Word Sense Disambiguation using Stacked Denoising Autoencoder	
<i>Kazuhei Kouno, Hiroyuki Shinnou, Minoru Sasaki and Kanako Komiya</i> .....	224
Construction of Semantic Collocation Bank Based on Semantic Dependency Parsing	
<i>Shijun Liu, Yanqiu Shao, Yu Ding and Lijuan Zheng</i> .....	232
Dynamic Semantics for Intensification and Epistemic Necessity: The Case of Yiding and Shibì in Mandarin Chinese	
<i>Jiun-Shiung Wu</i> .....	241
A Corpus-based Comparatively Study on the Semantic Features and Syntactic patterns of Yòu/Hái in Mandarin Chinese	
<i>Yuncui Zhang and Pengyuan Liu</i> .....	249
An Empirical Study on Sentiment Classification of Chinese Review using Word Embedding	
<i>Yiou Lin, Hang Lei, Jia Wu and Xiaoyu Li</i> .....	258
Polarity Classification of Short Product Reviews via Multiple Cluster-based SVM Classifiers	
<i>Jiaying Song, Yu He and Guohong Fu</i> .....	267
Automatic Classification of Spoken Languages using Diverse Acoustic Features	
<i>Yaakov Hacoheh-Kerner and Ruben Hagege</i> .....	275
The Syntactic and Semantic Analysis of Hěn X Constructions in Spoken Corpora	
<i>Yen-Ju Chen, Huei-Ling Lai and Shao-Chun Hsu</i> .....	286
Methods and Tool for Constructing Phonetically-Balanced Materials for Speech Perception Testing: A Development of Thai Sentence-Length Materials	
<i>Adirek Munthuli, Charturong Tantibundhit, Chutamanee Onsuwan and Krit Kosawat</i> .....	293
Graph Theoretic Features of the Adult Mental lexicon Predict Language Production in Mandarin: Clustering Coefficient	
<i>Karl Neergaard and Chu-Ren Huang</i> .....	302
Feature Reduction Using Ensemble Approach	
<i>Yingju Xia, Cuiqin Hou, Zhuoran Xu and Jun Sun</i> .....	309

Measuring Popularity of Machine-Generated Sentences Using Term Count, Document Frequency, and Dependency Language Model <i>Jong Myoung Kim, Hancheol Park, Young-Seob Jeong, Ho-Jin Choi, Gahgene Gweon and Jeong Hur</i> .....	319
Acquiring distributed representations for verb-object pairs by using word2vec <i>Miki Iwai, Takashi Ninomiya and Kyo Kageura</i> .....	328
Dependency parsing for Chinese long sentence: A second-stage main structure parsing method <i>Bo Li, Yunfei Long and Weiguang Qu</i> .....	337
A Light Rule-based Approach to English Subject-Verb Agreement Errors on the Third Person Singular Forms <i>Yuzhu Wang and Hai Zhao</i> .....	345
A Machine Learning Method to Distinguish Machine Translation from Human Translation <i>Yitong Li, Rui Wang and Hai Zhao</i> .....	354

# Identification of Sympathy in Free Conversation

**Tomotaka Fukuoka and Kiyooki Shirai**

Japan Advanced Institute of Science and Technology  
1-1, Asahidai, Nomi, Ishikawa, 923-1292, Japan  
{s1320010, kshirai}@jaist.ac.jp

## Abstract

Dialog systems are generally categorized into two types: task oriented and non task oriented systems. Recently, the study of non task oriented dialog systems or chat systems becomes more important since robotic pets or nursing care robots are paid much attention in our daily life. In this paper, as a fundamental technique in a chat system, we propose a method to identify if a speaker displays sympathy in his/her utterance. Our method is based on supervised machine learning. New features are proposed to train a classifier for identifying the sympathy in user's utterance. Results of our experiments show that the proposed features improve the F-measure by 3-4% over a baseline.

## 1 Introduction

Dialog systems could be broadly divided into two categories. One is a task oriented dialog system. It focuses on a specific task such as guidance on sight-seeing, hotel reservation or promotion of products, and communicates with a user to achieve a goal of the task. The other is a non task oriented dialog system or chat system. It does not suppose any specific tasks but can handle a wide variety of topics to freely chat with the user. Most of the past researches focus on task oriented dialog systems. In recent years, however, non task oriented dialog systems become more important since robotic pets or nursing care robots are paid much attention (Libin and Libin, 2004).

One of important characteristics in free conversation is sympathy of a speaker for the topics in

the conversation (Anderson and Keltner, 2002; Higashinaka et al., 2008). The topics in free conversation are not fixed but could be changed by the speakers at any time. To make the conversation natural and smooth, however, a non task oriented dialog system can not arbitrarily change the topics. It is uncomfortable for the user if the system would suddenly change the topic when the user wants to continue to talk on the current topic, or if the system would keep the same topic when the user is bored and does not want to talk on the topic any more. If the system fails to shift the topic at appropriate time, the user may break the conversation. The sympathy of the user is one of the useful clues to guess good timing for changing the topic. If the user shows the sympathy for the current topic, the system should continue the conversation with the same topic. On the other hand, if the user does not display the sympathy, the system should provide other topics. Therefore, it is essential for the chat system to guess the sympathy of the user.

This paper proposes a method to automatically judge whether the user displays the sympathy in his/her utterance as a fundamental technique in a non task oriented dialog system. In this paper, we define 'sympathetic utterance' as the utterance where the speaker expresses the sympathy or approval especially when he/she replies to subjective utterance of the other participant. Note that the utterance just showing agreement is not defined as sympathetic. Various kinds of clues could be applicable for identification of the sympathy, such as facial expressions, gesture or the contents of the utterance. Since we focus on a text based chat system, our

method only considers the content and detects the user’s sympathy in a transcript of the utterance. In addition to ordinary n-gram features, new features for the sympathy identification are introduced. The effectiveness of our proposed features will be proved via empirical evaluation.

The remaining parts of this paper are organized as follows. Section 2 discusses related work for the sympathy identification. Section 3 presents our proposed method. Section 4 reports results of evaluation experiments. Finally, Section 5 concludes the paper.

## 2 Related work

A considerable number of studies have been made on an automatic tagging of utterance in a dialog corpus. That is, each utterance in the dialog is automatically annotated with some useful information such as dialog acts. Hereafter we call it ‘dialog tag’. Supervised machine learning is often used for automatic identification of dialog tags. Since the sympathy of the speaker is also regarded as a kind of dialog tags, we introduce several related work automatically classifying utterance into dialog tags including the sympathy<sup>1</sup>.

Xiao et al. (2012) proposed a method to estimate the sympathy speech using the language model learning tool SRILM (Stolcke, 2002). In their method, n-gram of words were used as the features to classify if the utterance indicated the sympathy of the speaker. They reported that bi-gram was the most effective feature and the accuracy of the sympathy identification was around 60%.

A set of 29 dialog acts including ‘empathy’ was proposed toward an open-ended dialog system (Minami et al., 2012). They performed the automatic recognition of them using a weighted finite-state transducer with the words in the utterance.

Sekino et al. (2010) tried to identify the dialog acts using Conditional Random Fields (CRF). SWBD-DAMSL tag set (Jurafsky et al., 1997) were used as a set of dialog acts. Note that the tag ‘sympathy’ is included in SWBD-DAMSL. The features used for training CRF were the tag of the previous utterance, the number of content words in the utter-

<sup>1</sup>Since we focus on the methods that handle Japanese utterance, some of the related papers are written in Japanese.

ance, the length of the utterance and so on.

To identify the dialog acts of the sentences in microblogging, semantic category patterns were introduced as the feature of Support Vector Machine (SVM) classifier (Meguro et al., 2013). The words in the utterance were converted into their semantic categories (or abstract concepts) using a thesaurus, then n-gram of not words but semantic categories is used as the feature. Results of this study showed that n-gram of the semantic categories was more effective than word n-gram.

This study also applies supervised learning for automatic identification of the sympathy. Especially, we investigate what are the useful features to infer the sympathy in the utterance. Therefore, we focus on identification of the sympathy only, although many previous work handled the sympathy as one of the dialog acts. Several studies reported that characteristics of the sympathy could be found in an expression at the end of the utterance (Itoh and Nagata, 2007; Huifang, 2009). In addition, there might be more linguistic features indicating the sympathy of the speaker. The main contribution of the paper is that new features for the sympathy identification are proposed through manual analysis of a free conversation corpus. Furthermore, the effectiveness of these features is empirically evaluated by experiments. Note that the target language in this study is Japanese.

## 3 Proposed method

Our system accepts a text of utterance in free conversation as an input, then guesses whether it indicates the speaker’s sympathy. Support Vector Machine (SVM) (Chih-Chung and Chih-Jen, 2001) is applied to train a binary classifier to judge if the given utterance is sympathetic<sup>2</sup>.

### 3.1 Feature

We design the following 9 features for sympathy identification. Note that all features are binary, that is, the weight in the feature vector is 1 if it is present in the utterance, 0 otherwise.

#### $F_{ng}$ : Word n-gram

<sup>2</sup>Memory-based learning (TiMBL) (Daelemans et al., 2010) is also applied in our preliminary experiment, but SVM slightly outperformed TiMBL.

The word n-gram (n=1,2,3) is used as the feature, since it represents the content of the utterance. This is the basic feature widely used for identification of the dialog tags in the previous work. Since the content of the previous utterance is also important, we use the word n-gram of both the current and previous utterance.

**$F_{len}$ : Length of utterance**

Since the sympathetic utterance tends to be short, the length of the utterance (the number of characters) is considered. In the simple approach, the length feature is defined according to intervals, such as ‘1~5’, ‘6~10’ and ‘more than 10’. However, it is rather difficult to determine the optimum intervals. In this study, the length features are defined as in (1) and (2)

$$f_{len}^{(i)} : \text{ if } l_u \text{ is in } [i - 2, i + 2] \quad (1)$$

$$f_{len}^{(long)} : \text{ if } l_u \geq 20 \quad (2)$$

, where  $l_u$  stands for the length of the utterance. We use 17 length features  $f_{len}^{(i)}$  ( $3 \leq i \leq 19$ ) as well as an extra feature  $f_{len}^{(long)}$  indicating the utterance is long. This approach enables us to incorporate the information of the length of utterance into SVM more flexibly.

**$F_{tu}$ : Turn taking**

In our conversation corpus, the speakers may give two or more utterance in one turn. This feature indicates the presence of turn taking, i.e. whether the speaker of the current and previous utterance is the same.

**$F_{rw1}$ : Repetition of word (1)**

The speakers often show their sympathy by repeating a word in a previous utterance of the other. For example, in the simple conversation below <sup>3</sup>, the speaker B repeats the word ‘傑作 (fine work)’ to agree with A’s comment.

A: あの/ 映画 /は/ 傑作 /だ  
 (that) (movie) (fine work) (be)  
 (That movie is a fine work.)

B: 傑作 /だ/ね  
 (fine work) (be)  
 (It is a fine work.)

<sup>3</sup>Note that ‘/’ stands for the word segmentation, and a word or a sentence in parentheses is an English translation. The words without translations are function words that have no meaning.

We introduce a feature indicating if the same word appears in the current and previous utterance.

**$F_{rw2}$ : Repetition of word (2)**

Repetition of the words does not always indicates the sympathy. Let us consider the following example.

A: 海草類 / 嫌い /なの/?  
 (seaweed) (dislike)  
 (Do you dislike seaweed?)  
 B: そう/で/も/ない/よ、 / 海草  
 (so) (not) (seaweed)  
 (Not so much, seaweed.)

The speaker B repeats the word ‘海草 (seaweed)’, but his/her utterance does not show the sympathy.

This feature is similar to  $F_{rw1}$ , but more strictly checks the presence of repetition of the content words. The feature  $F_{rw2}$  is activated if either condition below is fulfilled:

- The last predicative word in the previous utterance is also found in the current utterance.
- There is only one content word in the current utterance and it also appears in the previous utterance.

**$F_{rc1}$ : Repetition of semantic class (1)**

Repetition of not words but semantic classes is considered in this feature. In the following example, no content word is overlapped in two utterance, but the speaker B express his/her sympathy by saying ‘楽しかった (fun)’ whose meaning is similar to ‘面白かった (interesting)’ in the speaker A’s utterance.

A: あの/ 映画 /は/ 面白かった /た  
 (that) (movie) (interesting)  
 (That movie was interesting.)  
 B: 楽しかった/た/ね  
 (fun)  
 (It was fun.)

This feature is activated if the same semantic class appears in both current and previous utterance. A Japanese thesaurus ‘Bunruigoihyo’ (National Institute for Japanese Language

and Linguistics, 2004) is used to obtain the semantic classes of the words. If one word has two or more semantic classes in the thesaurus, all of them are used to check repetition in two utterance. That is, we build the lists of all possible semantic classes of all content words in the current and previous utterance, and check if there is an overlap between them.

### $F_{rc2}$ : Repetition of semantic class (2)

Similar to  $F_{rw2}$ , repetition of the semantic classes are strictly checked as follows:

- The semantic class of the last predicative word in the previous utterance is also found in the current utterance.
- There is only one content word in the current utterance and its semantic class also appears in the previous utterance.

### $F_{da}$ : Dialog act

Dialog act is also a useful feature to identify the sympathy. When we hear the other’s assertion or opinion, we sometimes show our sympathy with it. However, we seldom express the sympathy for a simple yes-no question. In this study, we define a set of dialog acts in free conversation as in Figure 1.

self-disclosure,	question(yes-no),
question(what),	response(yes-no),
response(declarative),	backchannel,
filler, confirmation,	request

Figure 1: Dialog act

We manually annotate the conversation corpus with the dialog acts and use them as the features. In future, the dialog acts should be automatically identified to derive this feature.

### $F_{end}$ : End of sentence

The speakers often show their sympathy in an expression at the end of their utterance. For example, in Japanese, “だ [da] / ね [ne]” or “よ [yo] / ね [ne]”<sup>4</sup> at the end of the sentence strongly indicates the sympathetic mood of the

<sup>4</sup>Parenteses show pronunciation of each word. ‘/’ stands for word segmentation. Note that these words are particles and have no meaning.

speaker. Based on the above observation, the expression at the end is introduced as the feature. In this paper, it is represented by a sequence of function words at the end of each sentence in the utterance.

## 3.2 Combination features

In the preliminary experiment, we investigated several types of kernels of the SVM classifier: linear kernel, polynomial kernel, radial basis function and so on<sup>5</sup>. We found that the kernels except for the linear kernel performed very poorly on our data set. Therefore, we chose the linear kernel. However, the individual features are regarded as independent each other in the SVM with the linear kernel, although the dependency between the features should be considered.

To tackle this problem, we introduce a feature composed by combination of the existing features. When a feature set  $F = \{\dots f_i \dots\}$  is derived from one utterance, where  $f_i$  is one of the features described in Subsection 3.1, all possible pairs of features  $[f_i, f_j]$  ( $i \neq j$ ) are also added to the feature set. Hereafter,  $[f_i, f_j]$  is referred to as a combination feature. The combination features enable the classifier to consider the dependency between two features. Since the number of this feature are increased combinatorially, feature selection is applied as described in the next subsection.

## 3.3 Feature selection

A simple feature selection procedure is introduced. We apply the feature selection only for the word n-gram feature ( $F_{ng}$ ) and the combination feature, since the numbers of these features are extremely high.

The correlation between a sympathy class and a feature  $f_i$  is measured by  $\chi^2$  value. The features are discarded when  $\chi^2$  value is less than a threshold. We denote the threshold of  $\chi^2$  value for the n-gram and combination feature as  $T_{ng}$  and  $T_{comb}$ , respectively. In the experiment in Section 4, these thresholds will be optimized with a development data.

## 3.4 Filtering of negative samples

In supervised machine learning, it is inappropriate that the numbers of positive and negative samples in

<sup>5</sup>We used LIBSVM (Chih-Chung and Chih-Jen, 2001).

the training data are extremely imbalanced, since the trained classifier may display strong bias for the majority class. In general, however, the sympathetic utterance does not frequently appear in free conversation. Actually, the ratio of the sympathetic utterance is 1.1% in our conversation corpus as will be shown in Table 1. To tackle this problem, a filtering process to remove the negative samples is introduced to correct imbalance of the training data.

The basic idea of our filtering method is that we try to remove redundant negative samples. Here ‘redundant’ sample stands for a sample that is similar to other samples in the training data. Similar negative samples might be redundant and could be removed from the training data without any significant loss of the classification performance. The similarity between two samples (utterance) is measured by cosine similarity of the vector consisting of the word n-gram feature only.

It is time consuming to calculate the similarity between all possible pairs of the utterance in the training data. Instead, we reduce the computational cost by constructing clusters of the utterance as the prepossessing. First, the clusters are constructed from the set of the negative samples. A fast clustering algorithm ‘Repeated Bisections’ is used, where the number of the cluster is set to 1000<sup>6</sup>.

For each cluster, the redundant negative samples are detected by Algorithm 1. Given a set of utterance in a cluster  $U$ , the algorithm divides the utterance into a set of non-redundant utterance  $U_k$  to be kept and redundant utterance  $U_d$  to be deleted. For each utterance  $u_i$ , if the similarity between  $u_i$  and the rest of the utterance  $u_j$  is greater than the threshold  $S_{fil}$ ,  $u_j$  is added to the set  $U_d$ . Then  $u_i$  is added to  $U_k$ . Intuitively, if several similar utterance are found, only the first appeared one is remained in the training data. Note that the threshold  $S_{fil}$  controls the number of the removed negative samples. It is optimized on a development data.

## 4 Evaluation

This section reports experiments to evaluate our proposed method. In this experiment, the systems are evaluated and compared by the precision, recall and

<sup>6</sup>We used the clustering tool CLUTO. <http://glaros.dtc.umn.edu/gkhome/views/cluto>

**Input** :  $U = \{u_1, u_2, \dots, u_n\}$

**Output**:  $U_k, U_d$

$U_k \leftarrow \emptyset, U_d \leftarrow \emptyset$

**for**  $i \leftarrow 1$  **to**  $n$  **do**

**if**  $u_i \in U_d$  **then**

        | next

**end**

**for**  $j \leftarrow i + 1$  **to**  $n$  **do**

$sim \leftarrow \cos(u_i, u_j)$

**if**  $sim \geq S_{fil}$  **then**

            |  $U_d \leftarrow U_d \cup \{u_j\}$

**end**

**end**

$U_k \leftarrow U_k \cup \{u_i\}$

**end**

**Algorithm 1:** Search for redundant negative samples

F-measure of the identification of sympathetic utterance.

### 4.1 Data

Meidai conversation corpus<sup>7</sup> is used to train and evaluate our proposed method. It is a collection of transcription of actual conversation or chat in Japanese. Two to four participants joined free conversation. Dialogs where the number of the participants is two are chosen from the corpus, then each utterance is manually annotated with ‘sympathy tag’ indicating whether it expresses the sympathy of the speaker or not<sup>8</sup>.

We randomly divide the conversation corpus into three sets: 80% training, 10% development and 10% test set. Table 1 shows the number of the dialogs, sympathetic utterance (sym) and non-sympathetic utterance (non-sym) in each data. The ratio of the positive and negative samples stands at 1 to 86, that is, the number of sympathetic utterance is much fewer than non-sympathetic. A balanced data in-

<sup>7</sup><https://dbms.ninjal.ac.jp/nuc/index.php?mode=viewnuc>

<sup>8</sup>Each dialog in the corpus is annotated by one person. To measure inter-annotator agreement, another annotator put sympathy tags to only three dialogs. Cohen’s kappa was 0.27. It indicates the difficulty of the sympathy identification task. In future, the definition of sympathetic utterance should be more clarified to make a better annotation guideline for consistent annotation.

Table 1: Statistics in the conversation corpus

data	dialog	sym	non-sym
training	77	861	73378
development	10	103	8882
test	10	99	8598

cluding the same number of the positive and negative samples is also used for evaluation. It is made by keeping all positive samples and randomly choosing the equal number of the negative samples in the training, development and test data. We repeat to construct the balanced data five times, and evaluate the systems in these five data sets. Note that the results on the balanced data shown below are the average precision, recall and F-measure of five trials.

## 4.2 Results and discussion

### 4.2.1 Parameter optimization

First, the parameter  $T_{ng}$  for selection of n-gram feature was optimized on the development data. Figure 2 shows a change in precision(P), recall(R) and F-measure(F) on the development data. We chose  $T_{ng} = 0.9$  as the best parameter where the precision, recall and F-measure were the highest. In this case, 4378 features, which are 1% of all n-gram features, were selected.

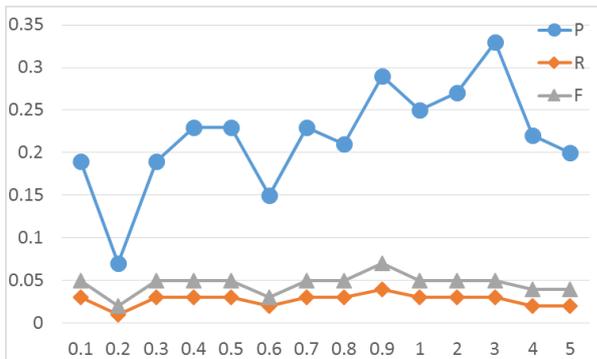


Figure 2: Optimization of  $T_{ng}$

Another parameters  $T_{comb}$  and  $S_{fil}$  were also optimized. The details will be reported later.

### 4.2.2 Results

We define the baseline as the classifier with the word n-gram feature only. Table 2 reveals the performance of the baseline and our proposed method

on the test data, while Table 3 shows the results on the balanced test data. In these tables, the filtering of the negative samples is not applied. Our proposed method outperformed the baseline on the whole, although the precision was comparable on the balanced data. In the imbalanced test data, the F-measure was not so high. This is because the sympathetic utterance does not frequently appear in the conversation corpus. Since the participants of some dialogs were strangers, they might hesitate to express their sympathy. On the other hand, in the balanced test data, the results were reasonably high. If our method is applied for the conversation between close friends where they frequently show their sympathy, it will achieve better performance than the results in Table 2.

Table 2: Results on the imbalanced test data

	P	R	F
Baseline ( $F_{ng}$ )	0.23	0.11	0.15
Proposed method	0.28	0.13	0.18

Table 3: Results on the balanced test data

	P	R	F
Baseline ( $F_{ng}$ )	0.80	0.73	0.76
Proposed method	0.81	0.76	0.80

### 4.2.3 Effectiveness of features

Next, to investigate the effectiveness of our proposed features, the models with several feature sets are compared. We train the classifiers with the basic word n-gram feature and one of the other features (denoted as  $F_{ng} + F_*$ ), and compared it with the baseline model ( $F_{ng}$ ). We also compare the classifier with all features (denoted as  $F_{ALL}$ ). Table 4 and 5 show the results on the imbalanced and balanced test data. Note that the combination features are not used in this experiment.

On the imbalanced test data, adding the feature  $F_{len}$ ,  $F_{rc2}$ ,  $F_{da}$  and  $F_{end}$  caused a decline of the F-measure. Furthermore, the classifier using all features were comparable with the baseline. However, on the balanced data, almost all types of the features contributed to gain the F-measure. In addition, precision, recall and F-measure of  $F_{ALL}$  were better than the baseline.

From the results in Table 4 and 5, turn taking ( $F_{tu}$ )

Table 4: Effectiveness of the features on the imbalanced test data

Feature set	P	R	F
$F_{ng}$	0.23	0.11	0.15
$F_{ng} + F_{len}$	0.18	0.08	0.11
$F_{ng} + F_{tu}$	0.25	0.12	0.16
$F_{ng} + F_{rw1}$	0.25	0.11	0.15
$F_{ng} + F_{rw2}$	0.26	0.11	0.15
$F_{ng} + F_{rc1}$	0.23	0.11	0.15
$F_{ng} + F_{rc2}$	0.21	0.10	0.14
$F_{ng} + F_{da}$	0.19	0.08	0.11
$F_{ng} + F_{end}$	0.19	0.10	0.13
$F_{ALL}$	0.24	0.11	0.15

Table 5: Effectiveness of the features on the balanced test data

Feature set	P	R	F
$F_{ng}$	0.80	0.73	0.76
$F_{ng} + F_{len}$	0.81	0.73	0.77
$F_{ng} + F_{tu}$	0.81	0.75	0.78
$F_{ng} + F_{rw1}$	0.81	0.73	0.77
$F_{ng} + F_{rw2}$	0.81	0.73	0.77
$F_{ng} + F_{rc1}$	0.81	0.72	0.76
$F_{ng} + F_{rc2}$	0.81	0.73	0.77
$F_{ng} + F_{da}$	0.81	0.73	0.77
$F_{ng} + F_{end}$	0.82	0.74	0.78
$F_{ALL}$	0.83	0.77	0.80

and repetition of word ( $F_{rw1}$  and  $F_{rw2}$ ) seem the most effective features. Since the increase or decrease caused by adding one feature is inconsistent for several features on the imbalanced and balanced data, however, the effectiveness of them are rather unclear.

#### 4.2.4 Effectiveness of combination feature

In this subsection, we evaluate the combination feature. Two sets of the features are investigated: the word n-gram feature  $F_{ng}$  and all proposed features  $F_{ALL}$ . For each feature set, the combination features are added to the feature vector of the utterance.

Recall that we introduce feature selection for the combination feature. The parameter  $T_{comb}$  was optimized on the development data.  $T_{comb}$  was set as 140 for both feature sets  $F_{ng}$  and  $F_{all}$  on the imbalanced data. While, it was set as 280 and 260 for  $F_{ng}$  and  $F_{all}$  on the balanced data, respectively.

Table 6 and 7 compare the classifiers with and without the combination feature in the imbalanced and balanced test data, respectively. In Table 6, the combination feature improves both precision and recall in  $F_{ALL}$  feature set. While, combination of the n-gram features increases the precision but decreases recall and F-measure. Therefore, the combination of our proposed features worked well, but the combination of n-gram not.

In the balanced test data (Table 7), the models with and without the combination feature are comparable.

Comparing  $F_{ng}+COMB$  and  $F_{ALL}+COMB$  in Table 6, incorporation of the proposed features improved the F-measure with a loss of the precision. In the same comparison in Table 7, all three criteria were improved by using the proposed features. Therefore, it can be concluded that our proposed features are effective for identification of the sympathy, especially when the dependency between two features is considered.

Table 6: Evaluation of the combination feature on the imbalanced test data

Feature Set	P	R	F
$F_{ng}$	0.23	0.11	0.15
$F_{ng}+COMB$	0.31	0.09	0.14
$F_{ALL}$	0.24	0.11	0.15
$F_{ALL}+COMB$	0.28	0.13	0.18

Table 7: Evaluation of the combination feature on the balanced test data

Feature Set	P	R	F
$F_{ng}$	0.80	0.73	0.76
$F_{ng}+COMB$	0.80	0.73	0.77
$F_{ALL}$	0.83	0.77	0.80
$F_{ALL}+COMB$	0.81	0.76	0.80

#### 4.2.5 Evaluation of filtering of negative samples

The method of negative sample filtering was evaluated using the imbalanced data set. First, the parameter  $S_{fil}$  was optimized as 0.5 that achieved the highest F-measure on the development data.

Three methods are compared in this experiment: a model without the negative sample filtering (w/o Filtering), a model with the filtering by our proposed method (Proposed Filtering) and a model where the

negative samples are randomly removed (Random Filtering). In Proposed Filtering, 25,174 negative samples were removed from the training data. In Random Filtering, the same number of the negative samples were randomly removed. We repeated the training of the classifier with random filtering five times and compared the average with the other methods.

Table 8: Evaluation of filtering methods

	P	R	F
w/o Filtering	0.28	0.13	0.18
Proposed Filtering	0.23	0.16	0.19
Random Filtering	0.25	0.18	0.22

Table 8 reveals the results of three methods. By the filtering, the recall was improved, while the precision declined. It is natural because the classifier tends to judge the utterance as sympathetic (positive) when the number of the negative samples in the training data is reduced. Since F-measure was improved, the filtering of the negative samples seems to contribute toward improvement of the performance. However, our proposed filtering method was worse than the random sampling. It is still uncertain why the idea to remove the redundant negative samples is inappropriate in this task. In future, we will investigate the reason and refine the algorithm of the negative sample filtering.

### 4.3 Error Analysis

We have conducted an error analysis to find major causes of the errors. First, we found many false positives (the sympathetic utterance is wrongly classified as non-sympathetic) and false negatives (the non-sympathetic utterance is wrongly classified as sympathetic) when the previous utterance was long. In such cases, the previous utterance consisted of many sentences, but only one sentence was usually related to the current utterance. Although many features were derived from the previous utterance, the most of them were irrelevant. Such noisy features might cause the classification error. To overcome this problem, the coherence between the current and previous utterance should be considered. In other words, it is required to introduce a method to choose only the sentence relevant to the current utterance

from long previous utterance.

Many errors were also found when both the current and previous utterance were too short. We guessed that the classification errors were caused by the lack of the features. Due to the feature selection, even the word n-gram feature was sometimes not extracted from short utterance. One of the solutions is to apply feature selection only for bi-gram and tri-gram while remaining all uni-gram features, in order to prevent from extracting no n-gram feature.

We also found that several false negatives were caused by the feature  $F_{end}$ . Some of the expressions at the end of the sentence indicate the speaker’s sympathy, but not always. Let us suppose such an expression appeared in non-sympathetic utterance and the lengths of both current and previous utterance were short. In such cases, since only a few features were extracted, the end of the sentence feature strongly worked to classify the utterance as the sympathetic. The way to incorporate the end expression into the classifier should be refined.

## 5 Conclusion

This paper proposed a method to identify the sympathetic utterance in the free conversation. The main contribution of the paper is to propose novel features for sympathy identification. Results of the experiments indicate that (1) the proposed features are effective, especially when the pairs of these features are considered as the additional features, (2) among the proposed features, turn taking and repetition of the content words show strong correlation with the sympathetic utterance, and (3) the filtering of negative samples is important to improve the F-measure.

F-measure of the proposed method was still low in the extremely imbalanced positive and negative sample data. We proposed the filtering method to remove the redundant negative samples, but it was worse than the random filtering. However, since the results on the balanced data were promising, we believe that the filtering of negative samples is a right way to improve the performance. In future, we will continue to explore a better way of negative sample filtering.

## References

- Alexander V. Libin and Elena V. Libin. 2004. Person-Robot Interactions From the Robopsychologists' Point of View: The Robotic Psychology and Robotherapy Approach. *Proceedings of the IEEE* 92, pp. 1789–1803.
- Ryuichiro Higashinaka, Kohji Dohsaka and Hideki Isozaki. 2008. Effects of self-disclosure and empathy in human-computer dialogue. *Spoken Language Technology Workshop*, pp. 109–112.
- Anderson, C. and Keltner, D. 2002. The role of empathy in the formation and maintenance of social bonds. *Behavioral and Brain Sciences* 25 (1), pp. 21–22.
- Bo Xiao, Dogan Can, Panayiotis G. Georgiou, David Atkins and Shrikanth S. Narayanan. 2012. Analyzing the Language of Therapist Empathy in Motivational Interview based Psychotherapy. *Proceedings of Asia-Pacific Signal and Information Processing Association Annual Summit and Conference*.
- Andreas Stolcke. 2002. SRILM – An Extensible Language Modeling Toolkit. *Proceedings of International Conference on Spoken Language Processing 2*, pp. 901–904.
- Yasuhiro Minami, Ryuichiro Higashinaka, Kohji Dohsaka, Toyomi Meguro, Akira Mori and Eisaku Maeda. 2012. POMDP Dialogue Control Based on Predictive Action Probability Obtained from Dialogue Act Trigram Sequence (in Japanese). *The Transactions of the Institute of Electronics, Information and Communication Engineers A*, Vol. 95, No. 1, pp. 2–15.
- Takahiro Sekino, Masashi Inoue. 2010. Tagging Extended Conversation Tag to Utterance (in Japanese). *Tohoku-Section Convention of Information Processing Society of Japan*, 10-6-B3-2.
- D. Jurafsky, E. Shriberg, D. Biasca. 1997. Switchboard SWBD-DAMSL Shallow-Discourse-Function Annotation Coders Manual Draft 13, University of Colorado, Institute of Cognitive Science, Tech. Rep, pp. 97-102.
- Toyomi Meguro, Ryuichiro Higashinaka, Hiroaki Sugiyama, Yasuhiro Minami. 2013. Dialogue act tagging for microblog utterances using semantic category patterns (in Japanese). *IPSJ SIG Technical Report*, Vol. 2013-SLP-98, No. 1, pp. 1–6.
- Huifang Zhang. 2009. The Semantic Type and Expression Function of YONE in Natural Dialogue (in Japanese). *TSUKUBA WORKING PAPERS IN LINGUISTICS*, No.2, pp. 17–32.
- Masako Itoh and Ryota Nagata. 2009. Rhetorical Function of a Sentence-Final Particle for Constructing Interaction in Discourse (in Japanese). *Cognitive studies: bulletin of the Japanese Cognitive Science Society*, Vol. 14, No. 3, pp. 282–291.
- Chang, Chih-Chung and Lin, Chih-Jen. C.-C. Chang and C.-J. Lin. 2001. LIBSVM: A library for support vector machines. Software available at <http://www.csie.ntu.edu.tw/~cjlin/libsvm>.
- Daelemans, W., Zavrel, J., Van der Sloot, K., and Van den Bosch, A. 2010. TiMBL: Tilburg Memory Based Learner, version 6.3, Reference Guide. Technical Report ILK 03-10, Tilburg University, ILK.
- National Institute for Japanese Language and Linguistics. 2004. Bunruigoihyo. Dainippon tosho.

# Toward a Corpus of Cantonese Verbal Comments and their Classification by Multi-dimensional Analysis

**Oi Yee Kwong**

Department of Translation  
The Chinese University of Hong Kong  
Shatin, N.T., Hong Kong  
oykwong@arts.cuhk.edu.hk

## Abstract

The information explosion in modern days across various media calls for effective opinion mining for timely digestion of public views and appropriate follow-up actions. Current studies on sentiment analysis have primarily focused on uncovering aspects like subjectivity, sentiment and credibility from written data, while spoken data are less addressed. This paper reports on our pilot work on constructing a corpus of Cantonese verbal comments and making use of multi-dimensional analysis to characterise different opinion types therein. Preliminary findings on the dimensions identified and their association with various communicative functions are presented, with an outlook on their potential application in subjectivity analysis and opinion classification.

## 1 Introduction

Nowadays there are numerous channels for expressing personal opinions. Views expressed in written forms are no longer confined to newspapers and magazines but are found everywhere in social media on the almost boundless internet. Meanwhile, the boom in all kinds of talk shows and phone-in programmes on radio and television have allowed both experts and non-experts to voice their views on many different

subjects such as politics, finance, entertainment and leisure, just to name a few examples.

The challenge in such information explosion has to be met by effective opinion mining, which has so far focused on the subjectivity, sentiment, credibility, etc. from written data. Spoken data and the sub-types of opinions are relatively less addressed, which motivated our current work.

Opinionated utterances, or verbal comments, are likely to form a specific informal spoken genre as social media text has made a specific type of written language. They are distinct for utterance lengths, incompleteness, presence of speech errors, self-repairs, and speech planning evidence, amongst others. The comments may also be further categorised according to their communicative functions, such as presenting the speaker's stance, giving advice to someone, providing information, making prediction, and evaluating or judging something. The effective classification of these different functions will be essential. This paper thus reports on a pilot study on the construction of a corpus of Cantonese verbal comments and the use of multi-dimensional analysis for characterising different types of opinions expressed therein, and discusses the potential application of the results in subsequent opinion mining work.

Section 2 reviews related work. Section 3 introduces our corpus of Cantonese verbal comments. Section 4 discusses the linguistic features used in the preliminary multi-dimensional analysis done in the current study and the initial results, while Section 5 concludes with future directions.

## 2 Related Work

Opinion mining often involves subjectivity and sentiment analysis. Subjectivity analysis aims at distinguishing opinionated sentences from factual statements, where the former is also known as private states, referring to one's mental and emotional states which may express one's attitude, feeling, beliefs, evaluation, speculation, etc. Sentiment analysis attempts to classify the polarity of subjective views as positive, neutral, or negative. A comprehensive survey can be found in Pang and Lee (2008), and Liu (2010).

Past studies have mostly been concerned with written data, typically first-hand opinions like movie reviews (e.g. Pang et al., 2002), product reviews (e.g. Hu and Liu, 2004), or debates on web forums (e.g. Somasundaran and Wiebe, 2009), and second-hand opinions reported or quoted in news articles (e.g. Wiebe and Wilson, 2002; Tsou et al., 2005; Ku et al., 2006).

Systems often leverage some sentiment lexicons (e.g. Wilson et al., 2005; Esuli and Sebastiani, 2006) and are thus primarily lexically based (e.g. Pang et al., 2002; Turney, 2002; Polanyi and Zaenen, 2006; Li et al., 2012), although tasks requiring more fine-grained information like opinion holders and targets would require more than simple lexical clues (e.g. Kim and Hovy, 2006; Lu et al., 2010; Zirn et al., 2011). Approaches using multi-lingual data are also gaining attention (e.g. Banea et al., 2010).

Subjectivity may be associated with various communicative functions, such as presenting one's stance, giving advice, making prediction, evaluating and commenting, etc. Such functions are achieved with a combination of rhetorical devices including but not limited to lexical choices. Corpus-based discourse analysis has thus often relied on multiple linguistic patterns to characterise register variations (Biber, 1988; Kaufer and Ishizaki, 2006).

Multi-dimensional analysis, as explained and applied in Biber (1988) as well as Conrad and Biber (2001), makes use of multivariate statistical techniques like factor analysis to identify salient linguistic co-occurrence patterns (called "dimensions") from a wide range of linguistic features. The dimensions are functionally interpreted and then used to characterise various spoken and written registers. Biber (1993), for

instance, identified five dimensions for the texts in the LOB corpus and London-Lund corpus based on 67 linguistic features. The first dimension has been labelled as "informational vs involved production", where the former is marked by features like word length, nominalizations, prepositions, etc. and the latter by present tense verbs, contractions, first and second person pronouns, etc.

The current work forms part of our project in which we investigate Cantonese verbal comments made in various domains and intend to use multi-dimensional analysis to characterise the comments and their respective communicative functions. Some preliminary results on corpus construction and the pilot study involving multi-dimensional analysis are reported and discussed in this paper. Our plan is to further employ the identified co-occurrence patterns of linguistic features for opinion mining in the future.

## 3 Corpus of Verbal Comments

### 3.1 Data Collection

The corpus compiled contains transcribed spoken Cantonese data from television and radio programmes broadcasted in Hong Kong during late 2013 to early 2014. They cover various domains (politics / current affairs, economics / finance, and food / entertainment) presented in different styles (such as interviews, phone-in programmes, singing contests, and food/film critics). Table 1 shows the data sources.

### 3.2 Pre-processing and Annotation

The transcription was done in verbatim with respect to individual speaker turns. The start time and end time for each turn were recorded. The role of a speaker within the programme (such as host, guest, reporter, and caller) was also noted. Self-repairs, hesitations, and pauses in the speech were indicated in the transcription accordingly. Table 2 shows an example, where the symbols //, ^^ and -- indicate intonational pause, self-repair and lengthening (by second) respectively. Transcription in Jyutping (for Cantonese) and an English translation for the content is given for reference. The talking speed for a given speech sample was calculated by the average number of syllables per minute.

Domain	Content and Programmes
Politics / Current Affairs	<p>Interview programmes on TV/radio by host(s) with a guest, sometimes with phone-in sessions</p> <ul style="list-style-type: none"> <li>• 星期六主場 (Face to Face): <i>A one-to-one interview programme on TV, produced by RTHK</i></li> <li>• 星期六問責 (Accountability): <i>An interview programme (with two hosts and one guest) broadcasted on radio, containing phone-in sessions, produced by RTHK</i></li> <li>• 講清講楚 (On the Record): <i>A one-to-one interview programme on TV, produced by TVB</i></li> </ul>
Economics / Finance	<p>TV programmes with discussions between host and financial analysts, sometimes with phone-in sessions</p> <ul style="list-style-type: none"> <li>• 理財博客 (Finance Blog): <i>A financial analysis programme on TV, usually with one host and one guest analyst, plus phone-in sessions, produced by ATV</i></li> <li>• 華爾街速遞 (Wall Street Express): <i>A financial analysis programme on TV, with one host and one guest analyst, containing phone-in sessions, produced by Cable TV</i></li> <li>• 樓盤傳真 (Property): <i>A real estate commentary programme on TV, with two or more hosts, reporters and interviewees, produced by Cable TV</i></li> </ul>
Food / Entertainment	<p>TV/radio programmes with critics on food/film, and singing contests on TV with judge comments</p> <ul style="list-style-type: none"> <li>• 一粒鐘真人蘇 (One Hour So): <i>An entertainment programme with a main host introducing food and restaurants with critics, sometimes with cooking demonstration, may have co-host in some episodes</i></li> <li>• 超級巨聲/星夢傳奇 (The Voice): <i>A series of singing contests on TV with instant comments from adjudicators, produced by TVB</i></li> <li>• 亞洲星光大道 (Asian Million Star): <i>A series of singing contests on TV with instant comments from adjudicators, produced by ATV</i></li> <li>• 電影兩面睇 (Movie World): <i>A film critics programme broadcasted on radio, usually with three hosts, produced by RTHK</i></li> </ul>

Table 1: Data Sources

Programme	星期六主場 (Face to Face)
Date	2013-10-12
Start time	00:02:09
End time	00:02:16
Role	Guest
Content	<p>呀 唔係 // 佢哋 倡議 嘅 嘢 冇 問題 // 但係  aa3 m4hai6 keoi5dei6 coeng3ji5 ge3 je5 mo5 man6tai4 daan6hai6  ah no they propose 's thing have-not problem but  Well, no, there is no problem with what they proposed, but</p> <p>佢哋 嘅 做法 呢 就 冇 問題 // 咁 所以 呢  keoi5dei6 ge3 zou6faat3 ne1 zau6 jau5 man6tai4 gam2 so2ji5 ne1  they 's method PAR ADV have problem so therefore PAR  the way they did it was problematic, and so</p> <p>即係 我就 覺得 -- 即係 要 ^ 要 出嚟 講  zik1hai6 ngo5 zau6 gok3dak1 zik1hai6 jiu3 jiu3 ceot1lai4 gong2  that is I ADV feel that is need to need to come out speak  I mean, I feel that ... well ... I have to ... have to speak out.</p>

Table 2: Example of a Speaker Turn

Sentiment annotation mainly follows the way explained in Wiebe et al. (2005). Annotators were asked to identify any opinions and other private states (e.g. beliefs, sentiment, speculation, etc.) expressed in the transcribed speech. First, the speech content may contain objective factual information as well as subjective opinions, as in the following examples respectively:

今集星期六主場嘅嘉賓呢//亦有參與其中  
(Our guest today on Face to Face also participated [in the event].)

即係你有理由呢 eh 我叫所謂越級偷步嘅  
(That means you have no reason to skip the steps and jump the gun.)

Second, opinions and other private states may be expressed explicitly with private state verbs or specific polar elements or implicitly with different styles of language, as in the following examples respectively:

呢個情況令人擔憂  
(This situation is worrying.)

咁終於都出咗呢個諮詢文件喇喺啱啱呢個  
禮拜啦  
(The consultation documents eventually come out this week. [implies the documents come out late])

Third, opinions may be from sources other than the speaker, especially when the speaker quotes someone else who is the source of the opinion. Fourth, opinions may be expressed with different strengths, which may have to be judged in context. Fifth, different attitudes may be conveyed in opinionated speech, which may typically be neutral, positive or negative. Sixth, the opinions or attitudes may be expressed with respect to certain things, people or events, which we call the target. Finally, opinionated speech may serve various communicative purposes, as demonstrated by the examples in Table 3.

For the annotation, the transcribed speech was first split into speech segments. In general the intonational pauses (marked with //) were taken as segment boundaries. Hence each speaker turn may contain one or more speech segments, and these

segments may make up one or more speech events. A speech event is considered to correspond roughly to a full sentence in written text.

For each speech event, as well as any private state expressed in a speech event, the following fields are to be filled: From (the starting segment), To (the ending segment), Word span (for events with explicit speech verbs), Subjective (whether it is an opinionated segment), Source (speaker by default or otherwise indicated), Target (the object of the opinion), Strength (how strong the opinion is: low, medium, high), Polarity (positive, neutral, negative), and Function (purpose of the speech event).

Polar elements, or expressions in the speech conveying positive or negative sentiments, are also identified. For each polar element, the following information is to be provided: From (the starting segment), To (the ending segment, usually the same as From), Word span (the expression conveying polarity), Source (speaker by default, otherwise indicate nested sources), Strength (how strong the expression is: low, medium, high), and Polarity (positive, neutral, negative).

Function	Example
<i>Stance</i>	業主唔應該再加租囉 (The owner should not further raise the rent.)
<i>Evaluation/ Comment</i>	唱就唱得唔錯//但台風差啲 (The singing is not bad, but the poise is not good enough.)
<i>Speculation</i>	我好懷疑呢單嘢係咪真 (I really doubt the truth of this case.)
<i>Prediction</i>	樓價應該會跌番啲 (Property prices will probably fall a little.)
<i>Elaboration/ Justification</i>	因為可能就嚟加息所以... (The interest rate will probably go up, therefore ...)

Table 3: Communicative Functions of Opinions

### 3.3 Materials for Pilot Study

The current pilot study made use of a subset of the corpus from two domains: current affairs and finance. Speaker turns by host or guest were selected. Only those turns which last at least 10

seconds were included. Each speaker turn was considered one speech sample. Drawn from about 150 minutes of transcribed speech from current affairs programmes and about 230 minutes of transcribed speech from finance programmes, a total of about 340 minutes of speech containing 495 samples with over 117K syllables were used in the analysis. The breakdown for individual domains and roles is shown in Table 4.

Domain	Time (mins)		Syllables		Samples	
	Host	Guest	Host	Guest	Host	Guest
Current Affairs	31.01	101.64	10,434	32,593	86	154
Finance	72.80	134.10	25,367	49,164	129	126

Table 4: Data Size for Current Study

## 4 Multi-dimensional Analysis

### 4.1 Linguistic Features Used

The following linguistic features were extracted and counted from the annotated materials described above. The quantitative data were then used in the current pilot analysis:

- Pronouns, including first person pronouns (我 *I*, 我哋 *we*), second person pronouns (你 *you*, 你哋 *you*), and third person pronouns (佢 *he/she*, 佢哋 *they*).
- Modals, including modal verbs likes 可能 *may*, 應該 *should*, 可以 *can*, 會 *will*, etc.
- Private verbs, including verbs indicating private states such as 諗 *think*, 覺得 *feel*, 認為 *think*, 相信 *believe*, etc.
- Yes/No question words, including 有冇 *have or have not*, 係咪 *did or did not*, etc. and other A-not-A patterns.
- Wh-question words, including *what* (乜, 咩, 嘢...), *why* (點解, 為乜...), *who* (邊個...), *when* (幾時, 如何...), and *how* (點樣...).
- Discourse connectives, including words indicating concession (雖然 *although*, 但係 *but*...), causal relation (因為 *because*, 所以 *therefore*...), conditions (除非 *unless*, 不論

*whether*...), and hypothetical situations (如果 *if*, 就算 *even if*...).

- Speech planning features, including common fillers like 即係 *that is*, 其實 *actually*, etc., as well as speed and number of self-repairs, hesitation, lengthening and pauses in a speech sample.

### 4.2 Procedures

The frequency data obtained for the above linguistic features were tabulated and subject to Factor Analysis, following the process of multi-dimensional analysis discussed in Biber (1988, 1993). Factor Analysis is a kind of multivariate analysis which reduces a large number of features to a smaller set of factors based on their co-occurring patterns. In this study, SPSS was used as the tool to do this. With reference to the factors identified and the loadings associated with their component features, dimensional scores were computed for each type of text (in this case, speech produced by a certain role in a certain type of programme) with respect to each factor/dimension. These scores were based on the average of the sum of normalized frequencies for positively loaded features less that for negatively loaded features under a particular dimension for a given text type.

### 4.3 Preliminary Results

Four factors, corresponding to the dimensions in multi-dimensional analysis, were identified in the process. As demonstrated by Biber (1993), each dimension could be functionally interpreted according to the positive features and negative features associated with it. For example, the abundance of personal pronouns, especially first and second person pronouns, might indicate a high degree of interaction and involvement. Individual text types, or genres, could be characterised by not just one but many features which often co-occur or are simultaneously absent. Given the relatively small set of features of limited variety used in this pilot study, not all dimensions were found to associate with negative features. The possible functional interpretations of the identified dimensions with the corresponding positive and negative features are shown in Table 5.

Dimension	Positive features	Negative features
D1: Interaction, Involvement, Stance	Private verbs 1st person pronouns Wh-questions 2nd person pronouns	
D2: Uncertainty, Prediction	Speed Speech fillers Modals Yes/No questions	
D3: Elaboration, Explanation	Speech planning features Causal connectives 1st person pronouns	2nd person pronouns Yes/No questions Concession words
D4: Argumentative	3rd person pronouns Concession connectives Hypothetical connectives Causal connectives	

Table 5: Dimensions Identified

The speech samples were divided into four categories (or registers) by the two domains (current affairs and finance) and two roles (host and guest). The dimensional scores for each category along each dimension were computed. Figure 1 shows the comparison of the categories along the first dimension (D1) and Figure 2 shows their comparison along the second dimension (D2). In the figures, “Cur” stands for current affairs and “Fin” stands for finance.

It can be seen that guests and hosts in both domains are quite clearly distinguished by D1, which is characterised by private verbs, first and second person pronouns, and wh-questions. These are indicative of interaction, involvement and stance. Guests in interview programmes are often asked for their views on certain subjects, while hosts are expected to remain as neutral as possible.

D2, which is characterised by faster speed and abundance of speech fillers, modals and yes/no questions, reflects the uncertainty of the speakers and is likely to be associated with predictions rather than factual statements. This dimension thus singles out guests in financial programmes, who usually make predictions on financial matters and give investment advice. Table 6 shows a guest speaker turn from each domain of similar duration for a quick visualisation of the features found for D1 and D2. The relevant features are bolded and underlined.

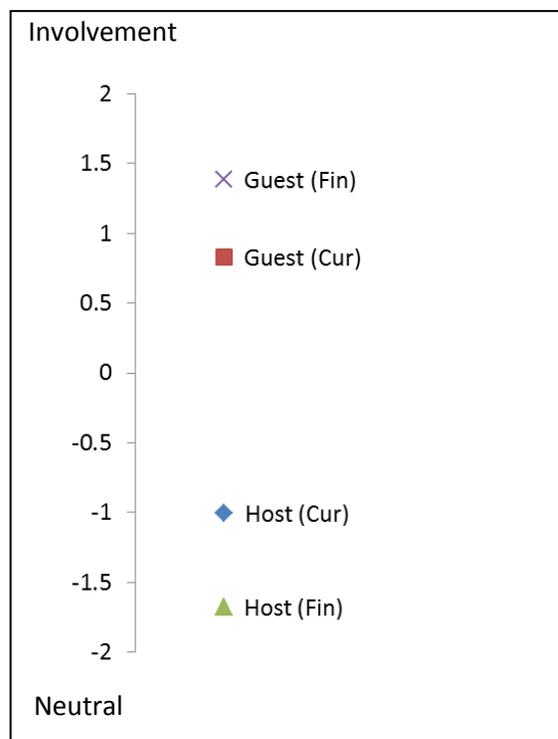


Figure 1: Comparison along D1

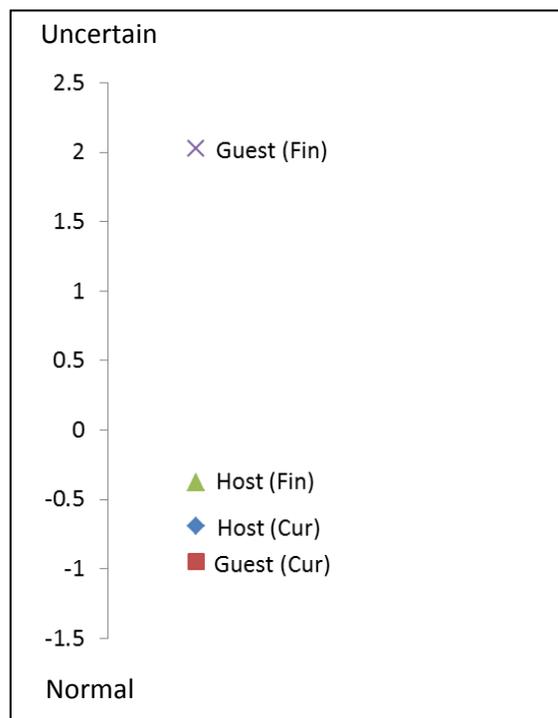


Figure 2: Comparison along D2

Dimension	Guest (Current affairs)	Guest (Finance)
D1	土地呢件事呢講親都好複雜嘅//因為有好多種--種類類嘅利益各個方面吓//一講親土地呢香港呢大家就即刻去^^向咗個錢字諗嘅//但係呢我哋其實講到土地嗰陣時一定係講全香港所有人嘅總體利益//嗱咁譬如話其實呢我哋人多嗰陣時點樣呢//就應該係已經有嘅城鎮一路周邊俾佢擴充出去呢//呢個係最好嘅方法嚟嘅 aha//咁就 eh 因此呢我哋冇理由走去山卡拉度起嘅樓㗎吓嘛	嗱汽車股嚟講嘅話呢我諗其實係暫時嚟講仍然會係比嗰個大市呢困擾住//咁但係整體嚟講嘅我覺得尤其是係長城啦或者係嗰個華晨呢//eh 長城食糊就靠 SUV 啦而華晨方面仍然係以佢嗰個比較高檔次嘅一啲 eh 豪華客車嚟講嘅話呢係受惠嘅//咁所以其實呢一類股份 eh..我 覺得就係逢低可以吸納啦//咁我 諗其實如果係嗰個跌定或者係內地嗰個銀根開始係寬^^放寬番嘅話呢//其實係可以吸納呢一類咁嘅股份囉
D2	土地呢件事呢講親都好複雜嘅//因為有好多種-種類類嘅利益各個方面吓//一講親土地呢香港呢大家就即刻去^^向咗個錢字諗嘅//但係呢我哋其實講到土地嗰陣時一定係講全香港所有人嘅總體利益//嗱咁譬如話其實呢我哋人多嗰陣時點樣呢//就應該係已經有嘅城鎮一路周邊俾佢擴充出去呢//呢個係最好嘅方法嚟嘅 aha//咁就 eh 因此呢我哋冇理由走去山卡拉度起嘅樓㗎吓嘛	嗱汽車股嚟講嘅話呢我諗其實係暫時嚟講仍然會係比嗰個大市呢困擾住//咁但係整體嚟講嘅我覺得尤其是係長城啦或者係嗰個華晨呢//eh 長城食糊就靠 SUV 啦而華晨方面仍然係以佢嗰個比較高檔次嘅一啲 eh 豪華客車嚟講嘅話呢係受惠嘅//咁所以其實呢一類股份 eh ..我覺得就係逢低可以吸納啦//咁我諗其實如果係嗰個跌定或者係內地嗰個銀根開始係寬^^放寬番嘅話呢//其實係可以吸納呢一類咁嘅股份囉

Table 6: Comparison on D1 and D2 Features

Figure 3 and Figure 4 show the comparison of the categories along the third and fourth dimensions (D3 and D4) respectively. D3, with the abundance of causal connectives and speech planning features, is characteristic of guests in both domains who often need to elaborate and explain the views, especially in current affairs discussions. Hosts in current affairs programmes often pose concise questions and let the guest respond, whereas those in financial programmes may pose more elaborated questions, or may even express some of their personal views with considerably more interaction with the guest. The differentiation of the categories along D4, for argumentation, suggests that guests tend to speak with more logical reasoning than hosts, and this is more evident for guests in financial programmes than those in current affairs programmes.

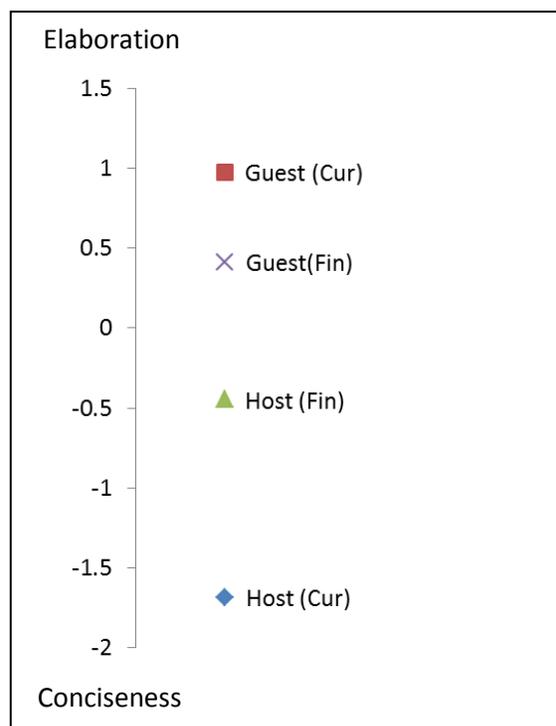


Figure 3: Comparison along D3

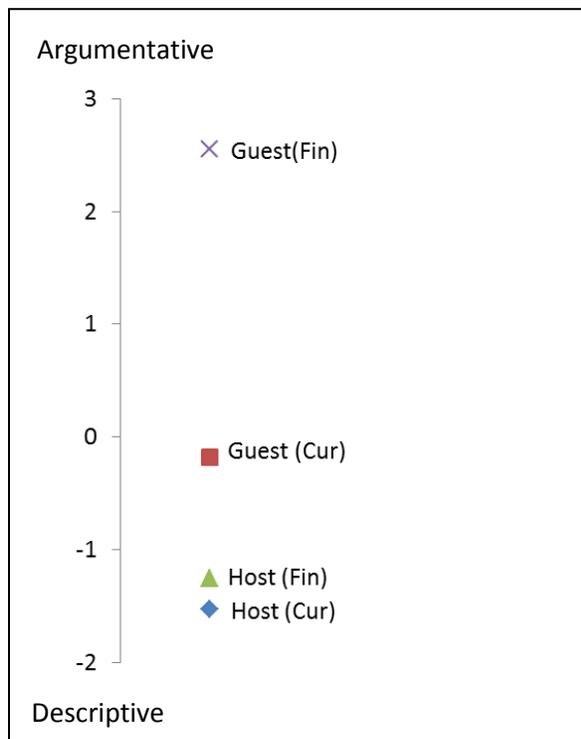


Figure 4: Comparison along D4

#### 4.4 Potential Application

Given the feature co-occurrence patterns identified and their association with specific communicative functions, they can potentially be used as the features in a variety of approaches, including those based on rules or machine learning, for automatic subjectivity recognition and opinion type classification. Larger-scale data annotation is in progress, for opinions and other private states. The resulting annotated corpus is expected to provide more data as well as more variety of linguistic features for a more comprehensive multi-dimensional analysis. The relevant features for characterizing different categories of verbal comments will be applied in experiments on opinion mining.

#### 5 Future Work and Conclusion

The preliminary study reported here suggested that multi-dimensional analysis is a promising approach to characterise opinionated speech samples and differentiate their sub-types based on communicative functions.

The immediate next step will expand the analysis to include more speech samples, possibly with a greater variety of domains and roles, to obtain more reliably distinguished dimensions, and to account for a wider range of opinion types and functions. So far we have relied mostly on lexical features, and more types of features will be necessary for a fuller picture of the genre characteristics of verbal comments. In particular, for lexical features we plan to add parts of speech, aspect markers, and sentence-final particles (which is very characteristic of Cantonese), and more importantly, for lexico-grammatical features we plan to include nominalisations, assertions, negation, and as far as possible, some discourse level features would be favoured. More tests on grouping and de-grouping the various features will be conducted and a more comprehensive analysis of the dimensions (with expanded datasets) will be done, for a descriptive account of Cantonese verbal comments as a specific spoken genre.

Another important direction will certainly be the application of the dimensions (and the features therein) and dimension scores for opinion mining. We have outlined their potential uses and experiments will be done when more annotated data for training and testing are ready. This future work is expected to showcase the synergy between corpus-based discourse analysis and opinion mining applications.

#### Reference

- Banea, C., R. Mihalcea and J. Wiebe. 2010. Multilingual Subjectivity: Are More Languages Better? In *Proceedings of the 23rd International Conference on Computational Linguistics (COLING 2010)*, Beijing, pp.28-36.
- Biber, D. 1988. *Variation across Speech and Writing*. Cambridge University Press.
- Biber, D. 1993. Using Register-Diversified Corpora for General Language Studies. *Computational Linguistics*, 19(2): 219-241.
- Conrad, S. and D. Biber. 2001. *Variation in English: Multi-Dimensional Studies*. Longman.
- Esuli, A. and F. Sebastiani. 2006. SENTIWORDNET: A Publicly Available Lexical Resource for Opinion Mining. In *Proceedings of the 5th Conference on Language Resources and Evaluation (LREC'06)*, pp.417-422.

- Hu, M. and Liu, B. 2004. Mining and Summarizing Customer Reviews. In *Proceedings of the ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)*, pp. 168-177.
- Kaufert, D. and S. Ishizaki. 2006. A corpus study of canned letters: mining the latent rhetorical proficiencies marketed to writers in a hurry and non-writers. *IEEE Transactions on Professional Communication*, 49(3): 254-266.
- Kim, S-M. and E. Hovy. 2006. Extracting Opinions, Opinion Holders, and Topics Expressed in Online News Media Text. In *Proceedings of the Workshop on Sentiment and Subjectivity in Text*, Sydney, pp.1-8.
- Ku, L-W., Y-T. Liang and H-H. Chen. 2006. Opinion extraction, summarization and tracking in news and blog corpora. In *Proceedings of AAAI*.
- Li, S., S. Ju, G. Zhou and X. Li. 2012. Active Learning for Imbalanced Sentiment Classification. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, Jeju Island, Korea, pp.139-148.
- Liu, B. 2010. Sentiment Analysis and Subjectivity. In N. Indurkha and F. J. Damerou (Eds.), *Handbook of Natural Language Processing*. Boca Raton, FL: Chapman & Hall.
- Lu, B., B.K. Tsou and T. Jiang. 2010. Supervised Approaches and Dependency Parsing for Chinese Opinion Analysis at NTCIR-8. In *Proceedings of the 8th NTCIR Workshop Meeting on Evaluation of Information Access Technologies: Information Retrieval, Question Answering, and Cross-Lingual Information Access*, Tokyo, Japan, pp.234-240.
- Pang, B. and L. Lee. 2008. Opinion Mining and Sentiment Analysis. *Foundations and Trends in Information Retrieval*, 2(1-2): 1-135.
- Pang, B., L. Lee and S. Vaithyanathan. 2002. Thumbs up? Sentiment Classification Using Machine Learning Techniques. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 79-86.
- Polanyi, L. and A. Zaenen. 2006. Contextual Valence Shifters. In J.G. Shanahan, Y. Qu and J. Wiebe (Eds.) *Computing Attitude and Affect in Text: Theory and Applications*. Springer.
- Somasundaran, S. and J. Wiebe. 2009. Recognizing Stances in Online Debates. In *Proceedings of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, Singapore, pp.226-234.
- Tsou, B.K., O.Y. Kwong, W.L. Wong and T.B.Y. Lai. 2005. Sentiment and Content Analysis of Chinese News Coverage. *International Journal of Computer Processing of Oriental Languages*, 18(2): 171-183.
- Turney, P. 2002. Thumbs Up or Thumbs Down? Semantic Orientation Applied to Unsupervised Classification of Reviews. In *Proceedings of ACL 2002*, Philadelphia, pp.417-424.
- Wiebe, J. and T. Wilson. 2002. Learning to disambiguate potentially subjective expressions. In *Proceedings of the Conference on Natural Language Learning (CoNLL)*, pp. 112-118.
- Wiebe, J., T. Wilson and C. Cardie. 2005. Annotating Expressions of Opinions and Emotions in Language. *Language Resources and Evaluation*, 39(2): 165-210.
- Wilson, T., J. Wiebe and P. Hoffmann. 2005. Recognizing Contextual Polarity in Phrase-level Sentiment Analysis. In *Proceedings of the Human Language Technology Conference and the Conference on Empirical Methods in Natural Language Processing (HLT/EMNLP)*, pp. 347-354.
- Zirn, C., M. Niepert, H. Stuckenschmidt and M. Strube. 2011. Fine-Grained Sentiment Analysis with Structural Features. In *Proceedings of the 5th International Joint Conference on Natural Language Processing*, Chiang Mai, Thailand, pp.336-344.

# Improved Entity Linking with User History and News Articles

Soyun Jeong, Youngmin Park, Sangwoo Kang, Jungyun Seo

Department of Computer Science and Engineering,

Sogang University,

Seoul, Korea

{soyun.j.nlp, pymnlp, gahng.sw}@gmail.com

seojy@sogang.ac.kr

## Abstract

Recent researches on EL(Entity Linking) have attempted to disambiguate entities by using a knowledge base to handle the semantic relatedness and up-to-date information. However, EL for tweets using a knowledge base, leads to poor disambiguation performance, because the data tend to address short and noisy contexts and current issues that are updated in real time. In this paper, we propose an approach to building an EL system that links ambiguous entities to the corresponding entries in a given knowledge base through the news articles and the user history. Using news articles, the system can overcome the problem of Wikipedia coverage, which does not handle issues in real time. In addition, because we assume that users post tweets related to their particular interests, our system can also be effectively applied to short tweet data through the user history. The experimental results show that our system achieves a precision of 67.7% and outperforms the EL methods that only use knowledge base for tweets.

## 1 Introduction

Recent development of the internet and computing technologies makes the amount of information increasing rapidly. Therefore, many long-term studies have been conducted on retrieving the needed information from the huge data. Named entity recognition(NER) and entity linking(EL) to specific entities as a part of information extraction now are actively attempt to extract meaningful knowledge in the huge information. The EL is the task of linking entity mentions in text to entities in a knowledge base.

As shown in Figure 1, the goal of entity linking is to map an ambiguous entity to its corresponding

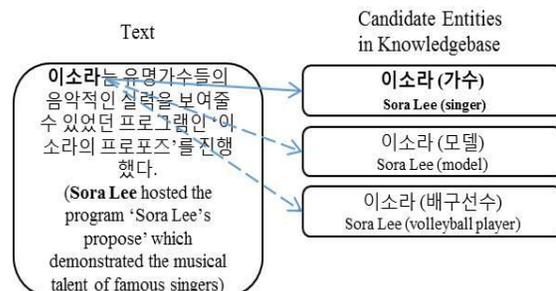


Figure 1: An example of entity linking. The bold type indicates an ambiguous named entity is in the text; the correct mapping entity is linked with the solid arrow

entity in knowledgebase. By leveraging the context information around an entity and knowledge base, ‘이소라 (Sora Lee)’ in the left box ‘Text’ in Figure 1 can be identified as the singer ‘이소라 (가수) (Sora Lee (singer))’. Context information can be a noun phrase; for example, ‘이소라의 프로포즈 (Sora Lee’s propose)’, which is the name of a music program hosted by ‘이소라 (가수) (Sora Lee (singer))’, can be known by knowledgebase.

Researchers have recently begun studying the problem of addressing named entities in informal and short texts. For example, Twitter, a popular microblogging platform, is updated and posted by users succinctly describing their current status within a limit of 140 characters. Java et al. (2007) showed that tweets address contents ranging from daily life to current events, news stories, and other interests. EL on Twitter has been used to identify entities from a structural knowledge base, e.g., Wikipedia, to enrich the task with additional features. To consider the characteristic of Twitter, state-of-the-art researches collectively link all entities in all tweets posted by a user via modeling the user’s interest (Shen et al., 2013; Bansal et al.,

2014). However, such methods cannot cover a EL for tweet task completely, because the posting the latest issues on tweet mentions, the most important characteristic of Twitter, cannot be applied.

In this paper, we first propose an EL method that considers Twitter contents addressing current issues and user interests through news articles and user history tweets besides knowledge base. In section 2, we describe recent EL studies. Section 3 provides our improved EL model with user history and news articles. Next, in section 4 we describe an experimental analysis in which we generated a Korean Twitter corpus and compared the contributions of each feature of our proposed method. Finally, we summarize our study with some concluding remarks in section 5.

## 2 Related Works

Traditional approaches have addressed the EL by dividing the task into two steps. The first step is NER, and the second step is entity disambiguation. Knowledge-based NER problem is different from the traditional NER. In the Traditional NER, while defining a class of such “PER” or “ORG” to entity, knowledge-based NER is to extract candidates of the fully qualified names of entities in knowledge base. For example, when recognizing the entity “이소라 (Sora Lee)”, a common NER classifies it into classes such as “PER”, while knowledge-based NER links it to specific entity such as “이소라 (모델) (Sora Lee (model))”. Early models of EL had tried a method of extracting only those corresponding to the named entity existing in knowledge base of all possible n-gram terms within document (Mihalcea and Csomai, 2007). Milne and Witten (2008) tried to utilize machine learning methods to recognize entities. Kim et al. (2014) uses hyperlinks within the Korean Wikipedia and a small amount of text manually annotated with entity information as training data. It employs a SVM model trained with character-based features to recognize entity. Liu et al. (2011) proposed an alternating two-step approach that alternates between the KNN classifier and CRF labeler in tweets. The KNN classifier models global features which span over long range of words. The CRF models the localized features among consecutive words. After recognizing entities in document, the next step is entity disambiguation. In section 2.1, we describe

previous works about entity disambiguation based on Wikipedia as knowledge base. Next, in section 2.2 describes state-of-the-art researches on EL via user modeling on tweets.

### 2.1 Entity Linking based on Wikipedia

Approaches leveraging Wikipedia for entity disambiguation started with Bunescu and Pasca (2006) and have been proposed in Cucerzan (2007), Han and Zhao (2009), Milne and Witten (2008), Charton et al. (2014). Bunescu and Pasca (2006) defined a semantic relatedness by similarity measure using Wikipedia categories. Later studies developed methods using richer structural features from the Wikipedia. The semantic relatedness is measured through the co-occurrence of links in Wikipedia articles. Milne and Witten (2008) have proposed to compute the mention to entity compatibility by leveraging the interdependence between EL. The system proposed that referent entity of a name mention should be coherent with its unambiguous contextual entities. Han and Zhao (2009) demonstrated how to leverage the semantic knowledge in Wikipedia, so the performance of named entity disambiguation can be enhanced by obtaining a more accurate similarity measure between name observations. Charton et al. (2014) built a representation of named entities that do not appear same as the knowledge base named entities.

### 2.2 Entity Linking via User Modeling

For EL on tweets aimed at short and noisy texts, the system should cover the insufficient context information contained in a tweet. To overcome such a problem, Shen et al. (2013) and Bansal et al. (2014) proposed an EL system via user modeling. Shen et al. (2013) suggested the KAURI system, a graph-based framework to collectively link all named entity mentions in all tweets posted by a user via modeling the user’s topics of interest. They assumed that each user has an underlying topic interest distribution over various named entities. Bansal et al. (2014) attempted to combine contextual and user models by analyzing a user’s tweeting behavior from previous tweets. This approach can be used for modeling users and disambiguating entities in other streaming documents. EL applied through user modeling systems outperforms systems using only a knowledge base.

In this paper, we adopted the traditional method that extracting only those corresponding to the named entity existing in knowledge base of all possible n-gram terms within document in the NER step. By proposing three models considering characteristics of the tweets, we focus on entity disambiguation step.

### 3 Entity Linking System with User History and News Articles

#### 3.1 Notation Framework

Our system is applied based on the user’s interest and current issues, and by considering which contents their Twitter mentions address. In this section, we introduce our proposed system, which consists of three scoring model systems, a Context modeling system, a User modeling system, and an Issue modeling system, as well the Linking model as shown in Figure 2. We adopted an existing method into the Context modeling system, by considering the context information around an ambiguous entity. The User modeling system uses the history mentions of user who posted the targeted mention. Targeted mention means the tweet mention with ambiguous entity, which we have to disambiguate. The Issue modeling system enriches the information from news articles, and can extract information that Wikipedia does not handle. The following section focuses on how our User modeling system works well in comparison to a Context modeling system. Furthermore we describe how the Issue modeling system improves the entire system to obtain a high level of performance.

- E – Target entity that should be linked
- $e^j$  – j-th non-ambiguous entity, which corresponds to an Wikipedia entity
- $c_j$ – j-th candidate entity for entity mention E that can be linked
- $\langle D \rangle$  – Sets of all entities in a document D
- $D_{c_j}^i$  – i-th news article that includes  $c_j$  as a topic,  $D_{c_j}$  means the set of all  $D_{c_j}^i$
- $[e]$  – Sets of all links in Wikipedia article whose title corresponds to entity e
- $S_c(c_j), S_u(c_j), S_i(c_j)$  – j-th candidate entity score of the Context modeling system, User modeling system, and Issue modeling system respectively

- Wikipedia article(page) title – Synonym of Wikipedia entity. Each article(page) in Wikipedia describes a specific entity and its title can be used to represent the entity it describes. Each article includes links which have semantic relation to its title. In other words, Wikipedia entities are considered to be semantic related if there are links between them.

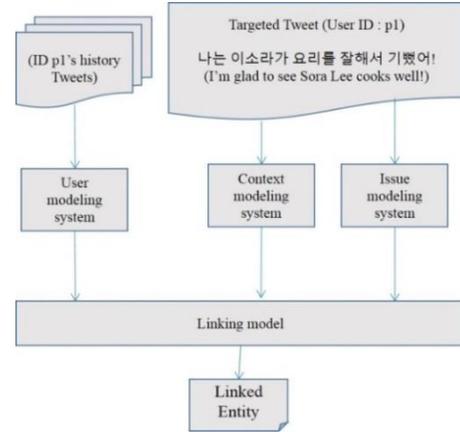


Figure 2: EL system

#### 3.2 Context Modeling System

The Context modeling system uses contextual information, meaning the non-ambiguous entities in a tweet including an ambiguous named entity. Most researches use this feature with Wikipedia category information, but we found that categories are often noisy(Milne and Witten, 2008) and that Korean Wikipedia pages provides insufficient category information compared with English Wikipedia pages. We therefore we adopted Eric Charton’s scoring method “mutual relation score” (Charton et al., 2014) without category information, as defined by the following formula :

$$S_c(c_j) = \partial dsr_{score}(e^j, c_j) + (1 - \partial)csr_{score}(e^j, c_j) \quad (1)$$

$$dsr_{score}(e^j, c_j) = |e^j \cap [c_j]| \quad (2)$$

$$csr_{score}(e^j, c_j) = \frac{|[e^j] \cap [c_j]|}{|[e^j]| + |[c_j]|} \quad (3)$$



Figure 3: Korean Wikipedia disambiguation page of named entity “이소라 (Sora Lee)” which appeared in user p1’s tweet mention shown on Figure 1

A Context modeling system fundamentally addresses the contextual features and links with a specific entity in Wikipedia. In Wikipedia, there is a “disambiguation page” that describes entities with the same name. As shown in Figure 3, the

disambiguation page for 이소라 (Sora Lee) lists three other people with the same name. The first 이소라 (Sora Lee) listed is a famous Korean model. The second 이소라 (Sora Lee) is a famous Korean singer. The last 이소라 (Sora Lee) is a member of the Korean national volleyball team. In this paper, we use the term  $S_c$  to indicate the “calculate score” in (Charton et al., 2014), which we use as our baseline system, as compared to systems with other added features.  $S_c$  implies simply exploiting a tweet as a context, not considering the properties of the tweet.

### 3.3 User Modeling System

As attributes of Twitter mentions, tweet contents range from daily life to current issues. Because the User modeling system understands the above property, it handles the user’s behaviors and interests. To address this concept, we utilize the past tweets of the user. We assumed that if a particular named entity is mentioned in a tweet, the user tends to have an interest in this named entity.

$$S_u(c_j) = \sum_{e^j \in \langle D \rangle} \partial dsr_{score}(e^j, c_j) + (1 - \partial) csr_{score}(e^j, c_j) \quad (4)$$

$$\langle D \rangle = \{e^j | j \text{ ranges from user's past tweets to the present tweet} \} \quad (5)$$

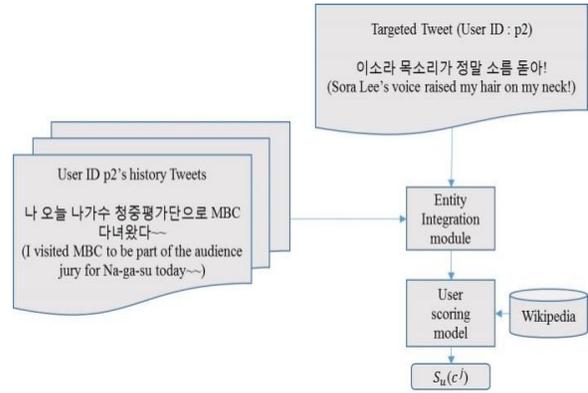


Figure 4 : User Modeling System

Figure 4 describes the process of the User modeling system. When the User modeling system detects a certain tweet of a user that includes an ambiguous entity, then we extract the user’s tweet history. The Entity Integration module then detects whether the feature  $e^j$  exists in Wikipedia entity by using the left-longest-match-preference method with *eojeol* uni-gram and bi-gram, then generate  $\langle D \rangle$ .  $j$  ranges from user’s past tweets to their last present tweet which to be disambiguated as shown in (5). We exploit the left-longest-match-preference method with *eojeol* uni-gram and bi-gram because tweet mentions tend to be grammatically incorrect and because in Korean, a noun always appears on the left-side in an *eojeol*, which consists of one or more morphemes comprising a spacing unit (Kang et al., 2014). Finally,  $S_u$  in the User scoring model is evaluated as in (4) and (5). In the example shown in Figure 4, the system has detected an ambiguous entity ‘이소라 (Sora Lee)’ in user p2’s tweet and extracted p2’s tweet history. The Entity Integration model then collect entities such as “나가수 (Na-ga-su)”, the TV program, and “MBC”, which is the broadcast station from p2’s tweet history. These features enhance the  $S_u(\text{이소라 (가수)})$ , because [이소라 (가수) (Sora Lee (singer))] includes “나가수 (Na-ga-su)” and “MBC”.

### 3.4 Issue Modeling System

The model described in section 3.3 has a disadvantage in that it cannot consider current issues, which is one of the characteristics of Twitter. A knowledge base focuses on major issues

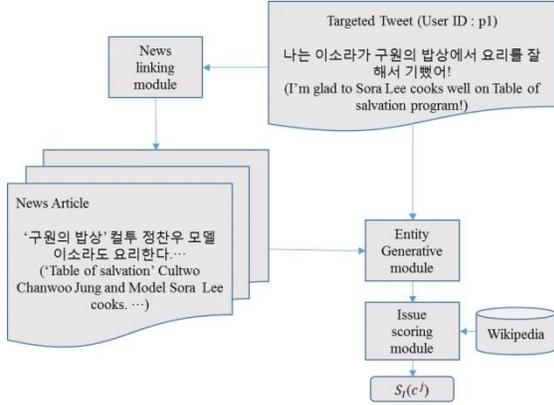


Figure 5: Issue Modeling System

and cannot provide all current issues in real time, for example, the recent events of a celebrity or trivial real-time events, which tend to be mentioned by Twitter users. Our Issue modeling system solves this problem by leveraging news articles. Because news articles address issues and events in real time, the Issue modeling system can extract current information.

As shown in Figure 5, when the Issue modeling system detects the ambiguous entity, “이소라 (Sora Lee)”, the news linking module extracts current news articles issued in within  $k$  days from the date of the tweet posted. The news articles have title included the detected entity  $E$ , 이소라 (Sora Lee). The module links the news articles to  $c_j$  by applying a cosine similarity between the article and Wikipedia page contents. In this example,  $c_1$ ,  $c_2$ , and  $c_3$  are “이소라 (가수) (Sora Lee (singer))”, “이소라 (모델) (Sora Lee (model))” and “이소라 (배구선수) (Sora Lee (volleyball player))” respectively. Accordingly, the Issue modeling system exploits the news articles as Wikipedia articles. Each article can be a  $D_{c_j}^i$ , which means the  $i$ -th news article which addresses  $c_j$  as a topic. The Entity Generative model generates the Wikipedia links in each news article. Single quotes in newspaper articles has the ability to display title or name. For example, the title of the book, the title of the movie, the title of the album and the title of the drama can be placed in single quotes[13]. Therefore we assume that news articles include important entities explicitly notated by punctuation. The Entity Generative model recognizes phrases or

words in single punctuation marks and nouns as entities, similar to Wikipedia links.

Table 1 describes the generation rules of the Entity Generative model and the example news article from Figure 5. Because the entity “구원의 밥상 (table of salvation)” is not actually included in the Wikipedia page on “이소라 (모델) (Sora Lee (model))”, it can be important information to link  $E$ , “이소라” to the correct answer,  $c_2$ , “이소라 (모델) (Sora Lee (model))”. The second generation rule, extracting entities that exist in Wikipedia article titles, the Entity generative model exploits a morpheme analyzer to extract nouns from news articles. This model uses a noun uni-gram and bi-gram to match the entity in Wikipedia article titles. Finally Issue scoring module determines the score  $S_i$ , as shown in (6).

$$S_i(c_j) = \frac{\sum_{i=0}^{|D_{c_j}|} \partial \text{dsr\_score}(e^j, D_{c_j}^i) + (1-\partial) \text{csr\_score}(e^j, D_{c_j}^i)}{|D_{c_j}|} \quad (6)$$

Generation Rule	Extracted Example
phrases or words in single punctuation marks and nouns	“구원의 밥상” (“table of salvation”)
entities that exist in Wikipedia article titles	“컬투”, “정찬우”, “모델” (“Cultwo”, “Chanwoo Jung”, “model”)

Table 1: Generation rule of the Entity Generative Model in Issue Modeling System

### 3.5 Linking Model

The Linking model finally disambiguates the entities based on the three models above, computes the Total Relatedness score,  $TR$ , and combines the scores  $S_c(c_j)$ ,  $S_u(c_j)$  and  $S_i(c_j)$  with parameter  $\alpha$ ,  $\beta$ , and  $\gamma$ , which are defined empirically. Equation (7) shows how this works.

$$TR(E, c_j) = \alpha S_c(c_j) + \beta S_u(c_j) + \gamma S_i(c_j) \quad (7)$$

$$(\alpha + \beta + \gamma = 1)$$

## 4 Experiments

### 4.1 Dataset and Framework

In the experiments, we evaluate our proposed method on the disambiguation of personal names, which is the most common type of named entity disambiguation. We created data set by collecting 50~60 tweets per 300 users who use twitter actively. Finally we collected 16,367 tweets in total. Then we selected tweets that contain the one person entity which exists in the list of entities in the Wikipedia’s disambiguation page. Finally we selected 248 tweets annotated same entity by 3 different annotators to verify reliability. The data set consists of 248 tweets including 248 disambiguous entities and they represent 33 ambiguous PERSON named entities. 33 ambiguous entities have 4.75 disambiguation pages on average in Wikipedia, 3.45 in data set. We conducted our experiments using  $k = 3$ . We defined  $\alpha, \beta$ , and  $\gamma$  empirically because they depend on the dataset. We used Korean Wikipedia as a knowledge base, the contents of which can be downloaded from <http://download.wikipedia.org/kowiki>. In our experiment, we dumped the latest Korean Wikipedia dump file, kowiki-2015-6-2-pages-articles. In the Issue modeling system, we adopted a Korean morpheme analyzer, “Jhannanum”<sup>1</sup>. In addition, we constructed a Wikipedia PER entity dictionary using the category information.

### 4.2 Experimental Result

Table 2 shows the performance of our proposed system. In the first row, the system is evaluated using only the Context modeling system. For the second row, we applied the User modeling system along with the Context modeling system. The system with the complete entity linking system obtained the results provided in the third row. We applied the accuracy score(number of true positives + number of true negatives/number of data set) to evaluate the system. We observe that the results of complete entity linking system are shown in the third row. We applied a precision score to evaluate the system. We observed that the complete algorithm provides the best results for

1. Semantic Web Research Center , JHannanum,  
<http://semanticweb.kaist.ac.kr/home/index.php/HanNanum>

our created test set. Considering that the 33 PER entities in our test set have 4.75 disambiguation pages on average, our proposed system performed well within a 67.7% level of accuracy. We also measured how precisely the Issue modeling system linked between news articles and Wikipedia pages. A 70.2% level of precision was achieved using only the cosine similarity. We showed that the complete system improves the performance of the Context modeling system when using user history and news articles.

System	Accuracy
<b>Baseline (Context Modeling System)</b>	31.5
<b>Baseline +User Modeling System</b>	58.9
<b>Baseline +User Modeling System +Issue Modeling System</b>	<b>67.7</b>

Table 2: Experimental Results

Table 3 shows the performance of News Linking module in Issue Modeling system. Ambiguous entity in collected news articles’s title were annotated by 2 different annotator. The Linking module performed within a 70.2% level of accuracy.

#news article	#entity type	# same name in news article	# same name in Wikipedia	Accuracy
836	20	1.4	3.35	0.72

Table 3: Performance of News Linking module in Issue Modeling system

Table 4 shows the extracted entities from each systems. First example shows improved performance by extracted entities from User Modeling System. This tweet user likes baseball as usual because he mentioned “삼성(Samsung)”, “롯데(Lotte)”, “야구장(ballpark)”, “조성환(sunghwan Cho)” in previous tweets. Among them, “삼성(Samsung)” and “롯데(Lotte)” appear in

Tweet	Extracted entities
@DooBoo_2 - 김태균이랑 동선이라닉ㅋㅋㅋㅋ (“Same line with Taegyun Kim kkk”)	Context modeling system
	“김태균(Taegyun Kim)”
	User modeling system
	“삼성(Samsung)”, “롯데(Lotte)”, “야구장(ballpark)”, “조성환(sunghwan Cho)” ...
@myhomenamsan - 어제 조인성을 봤다. 화장실에서 거울 봤는데, 우리가 엄마가 잘못했다 (“I saw Insung Cho yesterday. After I saw my face in the mirror. My mother’s fault.”)	Context modeling system
	“조인성(Insung Cho)”
	User modeling system
	“축구(soccer)”, “일본(Japan)”, “중국(China)” ...
	Issue modeling system
	“드라마(drama)”, “영화(movie)”, “SBS”, “배우(actor)”, “태국(Thailand)” ...

Table 4: Extracted entity examples from three modeling systems

“김태균(1971) (Taegyun Kim (1971))”. Further, entities extracted by Issue Modeling System enhance the system to resolve into “김태균(1971) (Taegyun Kim (1971))”. User Modeling system extracted entities in second example does not support the “조인성(배우) (Insung Cho(Actor))”. Instead, Issue Modeling system extracted “드라마(drama)”, “영화(movie)”, “SBS”, “배우(actor)” that support the system can link to “조인성(배우) (Insung Cho(Actor))”.

## 5 Conclusion

In this paper, we propose an entity linking system that, consists of three scoring model systems, a Context modeling system, a User modeling system and an Issue modeling system, along with a Linking model to integrate the three systems. We adopted an existing entity linking method as a baseline for Korean tweets, and by applying the User modeling system and Issue modeling system, it outperforms the baseline system, just using

knowledge base. Our system handles the characteristics of tweet mentions, such as current issues or trivial events that not described in a knowledge base, by using the User modeling system and Issue modeling system effectively.

However, because our work is the first to link entities with three different scoring model systems, the User modeling system does not use the additional features such as Twitter hashtag information. Furthermore, because only the left-longest-match-preference model and a noun unigram and bigram were used to detect entities in news articles in this experiment, the accuracy was not very high and should be improved later.

Further analysis is required for the user modeling and Issue modeling aspects of the system. Future work will also involve applying statistical methods to identify entities in news articles and using additional features appearing in Twitter for the User modeling system. Furthermore, we will adopt an efficient method to link news articles with tweets. We also plan to experiment on larger datasets and adopt our system to English tweets such as TAC\_KBP.

## Acknowledgment

This work was supported by the ICT R&D program of MSIP/IITP. [R0126-15-1112, Development of Media Application Framework based on Multi-modality which enables Personal Media Reconstruction]

## References

Akshay Java, Xiadan Song, Tim Finin and Belle Tseng. 2007. Why We Twitter: Understanding Microblogging Usage and Communities. *In Proceedings of the 9th WebKDD and 1st SNA-KDD 2007 workshop on Web mining and social network analysis*, 56-65.

Cucerzan Silviu. 2007. Large-Scale Named Entity Disambiguation Based on Wikipedia Data. *In Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, 7: 708–716.

David Milne and Ian H. Witten. 2008. *Learning to Link with Wikipedia*. *In Proceedings of the 18th conference on Information and knowledge management*, 215-224.

Donghyuk Lee. 2008. The Function of Single Quotation Marks on the Newspaper Articles. *Journal of Urimal*, (23):139-162.

Eric Charton, Marie-Jean Meurs, Ludovic Jean-Louis and Michel Gagnon. 2014. Mutual Disambiguation for Entity Linking. *In Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, 476–481.

- Rada Mihalcea and Andras Csomai. 2007. Wikify!: Linking Documents to Encyclopedic Knowledge. *In Proceedings of the 16th conference on Conference on information and knowledge management*, 233-242
- Razvan Bunescu and Marius Pasca. 2006. Using Encyclopedic Knowledge for Named entity Disambiguation. *In Proceedings of the 11th Conference of the European Chapter of the Association for Computational Linguistics*, 6: 9-16.
- Romil Bansal, Sandeep Panem., Manish Gupta and Vasudeva Varma. 2014. EDIUM: Improving Entity Disambiguation via User Modeling. *Journal of Advances in Information Retrieval*, 8416:418-423.
- Sangwoo Kang, Harksoo Kim, Hyun-Kyu Kang and Jungyun Seo. 2014. Lightweight morphological analysis model for smart home applications based on natural language interfaces. *International Journal of Distributed Sensor Networks*, 2014:1-9
- Wei Shen, Jianyong Wang, Ping Luo and Min Wang. 2013. Linking Named Entities in Tweets with Knowledge Base via User Interest Modeling. *In Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*, 68-76.
- Xianpei Han and Jun Zhao. 2009. Named Entity Disambiguation by Leveraging Wikipedia Semantic Knowledge. *In Proceedings of the 18th conference on Information and knowledge management*, 215-224.
- Xiaohua Liu, Ming Zhou, Xiangyang Zhou, Zhongyang Fu and Furu Wei. 2011. Recognizing Named Entities in Tweets. *In Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics*, 1:359-369.
- Youngsik Kim, Youngkun Hamn, Jisung Kim, Dosam Hwang and Ki-Sun Choi. 2014. A Non-morphological Approach for DBpedia URI Spotting within Korean Text, *In Proceedings of the 26th HCLT*, 100-106.

## Stance Classification on PTT Comments

**Ju-han Chuang**

Graduate Institute of Linguistics  
National Taiwan University  
r00142002@ntu.edu.tw

**Shukai Hsieh**

Graduate Institute of Linguistics  
National Taiwan University  
shukaihsieh@ntu.edu.tw

### Abstract

With the development of social media and online forums, users have grown accustomed to expressing their agreement and disagreement via short texts. Elements that reveal the user's stance or subjectivity thus becomes an important resource in identifying the user's position on a given topic. In the current study, we observe comments of an online bulletin board in Taiwan for how people express their stance when responding to other people's post in Chinese. A lexicon is built based on linguistic analysis and annotation of the data. We performed binary classification task using these linguistic features and was able to reach an average of 71 percent accuracy. A linguistic analysis on the confusion caused in the classification task is done for future work on better accuracy for such task.

### 1 Introduction

The wide spread of social media has given organizations and individuals new channels to understanding public opinion. Opinions are expressed via public debate forums and on various platforms, such as Facebook, Twitter, and even Youtube. These opinions reveal how users feel about an event, a person, or any focus of discussion. One expression in Taiwan, 測風向 *cè fēngxiàng* "to test the direction of the wind" is used by netizens when an article online inquires how the public feels about a topic. The phrase perfectly demonstrates how online discussions reflect the public's reaction to certain event or certain individual, often a political figure. In Taiwan, the mass media often resort to online forums as a source of understanding how the public responds

to political events like new policy and candidates running for elections.

Online discussion forums and social media give citizens an easier access to information and more power in shaping what information or idea gets passed on. Users of these online forums participate in a process of framing discussions and forming opinions. As Walker et al. (2012) pointed out, these debates involve not only the expression of opinions but also the formation of opinions. Through posting articles online, users talk about their beliefs on what is true or not, what is important, and what should be done. Their shared opinions thus stimulate more discussions. These users play an important role on how the discussions are framed and shape the form of the arguments.

One characteristics of these forums is that users usually have to express their position in a very short text. This implies that stance classification on short text would be different from identifying stance on a document level. Thus, we find it important to identify "elements" that reveal user's subjectivity in these short texts. Such resources would assist in identification or classification of attitudes and is applicable in all tasks that involves differentiating between factual information and opinionated utterances.

In the current study, we observe stance-taking language and arguing behavior from online comments and from previous studies in both English and Mandarin. The hope is to provide linguistic patterns and analysis that would assist in automated classification on stance. In the following sections, we will introduce previous works done on related topic of interest, discuss our work on tagging and classifying PTT comments, present the result of our classification task, and an analysis on classification errors that could shed light on future tasks on short text stance classification.

## 2 Related work

The importance of social media has been captured in Shirky's study on the political power of social media. He asserts that regular citizens, nongovernmental organizations, firms, and governments are all actors in social media. Social media has become an active part in political movements all over the world (Shirky, 2011).

This increasing importance and the accessibility of online data have triggered interests in related research to achieve automated methods in understanding affections and opinions. Previous research has made efforts on differentiating factual information from opinionated information. Opinionated information reveals a person's private states through the use of subjective language. Private state is a term that covers a person's overall attitude, including opinions, evaluations, emotions, and speculations (Quirk et al., 1985). Identifying these cues could assist automatic tasks on detecting attitudes online by providing resources (Wiebe et al., 2005).

Wiebe et al. (2004) extracted subjective cues by combining manually annotated subjective elements and expanding it with collocations and clustering method. Somasundaran et al. (2007) inspected dialogues in meetings to detect arguing and sentiment. In the annotated data, sentiment includes emotions, evaluations, judgements, feelings, and stances. Arguing refers to cues that indicate the speaker's attempt to convince one another.

The extracted subjective cues are utilized in classification of texts online for users' stance, defined as "an overall position held by a person toward an object, idea, or proposition" (Somasundaran et al., 2009). Stance classification deals with two sided debates and seeks an automated approach to categorization whether a person is for or against the topic discussed (Hasan and Ng, 2013).

In Somasundaran and Wiebe's study in 2010, they tested a combined feature set of arguing based features and sentiment based features. Arguing based features included arguing trigger expressions and modal verbs. Sentiment lexicon compiled by Wilson et al. in 2005 was used as sentiment based features. They reached an average accuracy of 64 percent classifying online debates based on the lexicon.

Anand et al. (2011) combined the feature set with metalinguistic features like word length and number of characters and approach arguing language with dependency parsers that capture words and its modifying targets. An average accuracy of 65 was reached. Hasan and Ng (2013) takes into account features like the author's position towards other issues and the stance of the immediate preceding post as predictors for stance classification and raised the accuracy up to 74%.

Faulkner (2014) incorporated generalized stance proposition subtrees and "Wikipedia Link-based Measure" to capture the relations between topics. The combined feature set was able to achieve an average accuracy of 80 percent on students' argumentative essays.

Although previous studies on stance classification has proven that classifier trained on unigrams could be a baseline that is hard to defeat and that identification on stance could be difficult for human annotators, adjustment according to the nature of the data set could help improve the results of the classification. Previous studies have mainly focused on document-level stance (Faulkner, 2014) or online debate forums (Anand et al., 2011; Hasan and Ng, 2013). Less attention has been placed on short text comments. However, we believe subjective elements is important in these texts full of sarcasm, typing errors, and colorful use of language (Malouf and Mullen, 2008). The aim of the current study is to establish related resources in Mandarin from short text comments online and to examine whether these linguistic cues assist in stance classification.

## 3 Methodology

### 3.1 Data collection

The corpus in the current study was collected from an online forum used in Taiwan, PTT. PTT is the most popular online bulletin board in Taiwan (Shea, 2006). It allows users to share their opinion by posting articles and responding to other's posts. The platform is divided into boards with different topics. Each board is centered on certain field of discussion. For example, the board "Boy\_Girl" is a board users discuss relationships between boys and girls.

In PTT, users give response to other users' posts with comments. Comments are tagged by the users

with their own attitude, whether towards the issue discussed, the author of the post, or previous responses comments left by other users. Three tags are available, including “push”, “boo”, and “arrow”. “Push” indicates that the author has a positive attitude towards either the original post or previous comments; “boo” is used when expressing a negative or opposing view; “arrow” is used when no certain attitude is chosen.

The data collected are extracted from three boards that are popular on PTT, including “Gossiping”, “Boy\_Girl”, and “WomenTalk”. The boards are chosen with consideration to the amount of data and to the nature of discussion. Some of the boards, though popular enough, may only allow “push” and “arrow” comments or may not be discussion-oriented. In order to identify the patterns used in push comments and boo comments, boards with more opinionated discussions are preferred.

Each line of comment in PTT is limited to 27 Chinese characters. Comments that exceeds 27 characters would be shown in a second line with an automatically assigned “arrow” at the beginning of the line. As a result, for comments that lasts more than one line, only the beginning line would be shown with the original tag while the rest of the lines would begin with an arrow. Since comments are extracted line by line with its tag at the beginning of each line and categorized as such, we cannot distinguish comments tagged with “arrows” by the original user from comments that exceeds one line. In order to avoid confusion between opinionated comments over one line and neutral comments that are originally tagged with arrows, the current study extracts only comments that are tagged with “push” and “boo” and focus on binary classification on opinionated sentences. Table 1 shows the details of the corpus used in the study.

		Number of comments	Number of tokens	Number of token types
Gossiping (6 months)	Push	3786034	28341656	11538420
	Boo	1222735	9000728	493926
Boy_Girl (12 months)	Push	998327	10006638	462780
	Boo	53376	508778	66186
WomenTalk (12 months)	Push	167473	1655771	121794
	Boo	36381	354672	47904

Table 1. Corpus information

### 3.2 Annotation criteria

Since comments on these forums are used as a way for users to express their opinion, to oppose to others’ ideas, and to justify their reasons for believing in or not believing in something (Wilson and Wiebe, 2005; Wilson, 2008; Somasundaran et al., 2007), the lexicon used in the classifier is compiled with a set of categories that are related to stance-taking and arguing. Following previous studies, we look for linguistic cues that indicate the author’s opinion or position on the discussed topic. The following are categories included in the annotation. In the tagging process, the identified “element” is not restricted to the word level. Considering the fact that subjectivity is often revealed in a common phrase or expression, function words are also included in the tagged set. For example, expression like 最好是 *zuihǎoshi*

“it’d better be” is treated as an element used to reject other people’s opinion.

#### 3.2.1 Arguing cues

Phrases and syntactic patterns that are indicative of opinionated sentences are manually identified from 5000 random comments tagged with push and boo, individually. Reynolds and Wang’s (2014) categorized comments on PTT into 9 categories, including *questions*, *reply*, *clarification*, *interpretation*, etc. We narrowed the categories down to 6 categories, including *question-answering*, *confirmation*, *counterargument*, *clarification*, *suggestion*, and *encouragement*. Expressions that carry one of these six functions would be included as an arguing cue. The annotated outcome is combined with the sixteen categories of arguing cues in MPQA opinion corpus (Wiebe et al., 2005; Somasundaran et al., 2007) as features for arguing cues.

Neutral question answering usually happens when users enquire information on something and is often non-opinionated. It often contains only a proper noun and with no specific cues. Sometimes users would include example-giving as part of their answer. Markers used at such circumstances would include phrase like 像是 *xiàngshì* “like”. Confirmation contains expressions used to agree with previous propositions, such as 同意 *tóngyì* “agree”. Counterargument is used when the user opposes to or challenges either the original post or previous comments. An example cue of counterargument would be 你怎麼知道 *nǐ zěnmē zhīdào* “how do you know”. Clarification is used when the focus of the comments shifts from one part to another and is sometimes used for similar purpose as a counterargument. An example of a comment used to clarify is shown in example (1) below, with the arguing cue underlined.

- (1) 你 爸爸 這樣是  
*nǐ bàba zhèyàngshì*  
 your father this is  
 錢奴 , 不是  
*qián nú, bùshì*  
 miser not  
 企業家 ...  
*qǐyèjiā*  
 entrepreneur  
 “Your father is a miser, not an entrepreneur.”

Suggestion is used when the user provides a solution or advice for the poster or other users. It is similar to neutral question answering but it usually involves more personal point of view. A typical cue in this category would be 建議 *jiànyì* “suggest”. Encouragement refers to the expressions of sympathy and support, which is very common on some boards. Users may use cues like 拍拍 *pāipāi* “patting” or 加油 *jiāyóu* “cheer up” to show their understanding of what the poster is going through.

### 3.2.2 Subjective elements

Following previous studies, words that are indicative of the author’s stance on the discussed topic are included in the lexicon. Our definition of subjective elements is similar to the one brought up by Wiebe in 1994, which identifies a subjective element as an element that is potentially subjective,

meaning that it can subjective in a certain context. Most of the words included are noun phrases and verb phrases that are evaluative, including both *explicit subjective elements* and *expressive subjective elements* (Wiebe et al., 2005; Wilson, 2008). Criticism and appraisal are given as tags to each of the phrases, indicating positive and negative evaluation.

*Explicit subjective elements* refer to phrases that explicitly show the attitude of the speaker, such as 討厭 *tǎoyàn* “hate” and 反對 *fǎnduì* “against”. *Expressive subjective elements* refer to expressions that reveal one’s attitude without explicitly naming that attitude. For example, in the sentence “the report is **full of absurdities**”, the phrase full of absurdities is used to express negative evaluation on the report (Wiebe et al., 2005).

In this category, *expressive subjective elements* are considered more interesting because some of the words might not be negative when it occurs individually or in other contexts. However, users on PTT form their habitual use of language to express their attitudes towards something without directly giving an evaluation. For example, the original definition of the word 公主 *gōngzhǔ* “princess” refers to a member in the royal family, but in PTT, it is a negative evaluation which refers to girls who rely on their boyfriends to take perfect care of them, cater to their every need, and gets mad over trivial matters. These expressions involve users’ world knowledge and is often used in sarcasm and irony (Wiebe et al., 2005). Identifying these elements would help us identify whether a comment or an evaluation towards the posted article contains positive or negative attitude.

### 3.2.3 Metadiscourse markers

Metadiscourse has been included in previous studies (Vande Kopple, 1985; Hyland, 1998; Hyland, 2002; Hyland and Tse, 2004; Dafouz-Milne, 2008) as a crucial part of persuasive writing. It reveals the author’s strategic arrangement of the text base on his intention to persuade and his understanding of the potential readers. According to Halliday (1973), the three macrofunctions of language include ideational function, interpersonal function, and textual function. The categorization of metadiscourse markers corresponds to two of the three functions, interpersonal and textual. Textual metadiscourse refers to the structure of the

text. How the author arranges his text might affect the readability persuasiveness of the text. Interpersonal metadiscourse, on the other hand, refers to how the author positions himself in the text and how he includes his readers. Following Hyland’s study (1998), ten categories are included

as metadiscourse markers: *logical connectives, frame markers, endophoric markers, evidentials, code glosses, hedges, emphatics, attitude markers, relational markers, and person markers*. Examples and definition of each category is given in the following table.

Textual Metadiscourse		
Logical connectives	Express semantic relation between main clauses	所以 <i>suǒyǐ</i> ‘therefore’
Frame markers	Explicitly refer to discourse acts or text stages	先 <i>xiān</i> ‘first’
Endophoric markers	Refer to information in other parts of the text	我剛才說的 <i>wǒ gāngcái shuō de</i> ‘what I just said’
Evidentials	Refer to source of information from other texts	指出 <i>zhīchū</i> ‘pointed out (in the show)’
Code glosses	Help readers grasp meanings of identical material	換言之 <i>huànyánzhī</i> ‘in other words’
Interpersonal Metadiscourse		
Hedges	Withhold writer’s full commitment to statements	可能 <i>kěnéng</i> ‘possibly’
Emphatics	Emphasize force or writer’s certainty in message	絕對 <i>juéduì</i> ‘definitely’
Attitude markers	Express writer’s attitude to propositional content	同意 <i>tóngyì</i> ‘agree’
Relational markers	Explicitly refer to or build relationship with reader	你 <i>nǐ</i> ‘you’
Person markers	Explicit reference to author(s)	我們 <i>wǒmen</i> ‘we’

Table 2. Categories of Metadiscourse Markers

During the annotation process, we find that there may be overlapping categories for arguing and for metadiscourse. One element could also have more than one function in comments. Our approach is to keep all categorization as part of the resources. Examples showing the arrangement of the data can be found in Table 3. The second column shows its category in metadiscourse, and the third column shows its category in MPQA arguing lexicon. The

fourth column shows its category in the six types of comments. The fifth column shows its annotated prior subjectivity, which is the polarity of the word when it stands alone. The last column show its polarity in the extracted corpus, which is acquired by comparing the element’s relative frequency in push and boo comments. The combined annotated lexicon includes a total of 4582 entries.

entry	metadiscourse	arguing	commenting	prior sub	calculated sub
感覺		assessment	question-answering; encouragement	neu	pos
不然	logical connectives	conditional	counterargument; suggestion	neu	pos
當然	emphatics	emphasis	confirmation; counterargument	neu	pos
好像	hedges		counterargument	neu	pos

Table 3. Examples of the subjective lexicon

### 3.3 Building the classifier

The combined lexicon is used as feature set for

identifying the stance of comments. In the current study, three sets of features are used in building the classifier. The first set of features contains subjective elements acquired through

manually annotating the data. For subjective elements, we assume negative evaluation reflects negative attitudes that more likely occur in boo comments while positive evaluation is associated with push comments. The second set of features includes the C-LIWC wordlist of positive and negative emotions (Huang, 2012). In this set, positive emotion words are associated with push comments while negative emotions are associated with boo comments. As for the rest of the cues, which may occur in both positive and negative context, we use relative frequency as a way of deciding whether it is representative for a certain position or not. Using the following calculations, if the number is higher than 0.70, the expression (which could be a subjective element, an entity, or even a disclaimer) would be judged as a feature for identifying that particular stance.

$$\frac{\text{Relative frequency of the segment in boo/push comments}}{\text{Relative frequency of the segment in all comments}}$$

The calculation is done after all of the scarce words are removed from the data. We used the third quantile of frequency as the threshold for scarce words. Thus, in all three sets of data, words that occur only once are removed. Relative frequency of data from each board is calculated individually. The combined wordlist is then used as features for an SVM classifier<sup>1</sup>.

#### 4 Result and discussion

In order to make a comparison, a baseline was done using segmented words as features for the SVM classifier. The feature set raises the accuracy on WomenTalk from 55 percent to 75 percent. The classification on Boy\_Girl data also improved by 13 percent. What's worth noticing is that the

<sup>1</sup> The classifier used here is released by CLiPS, Computational Linguistics and Psycholinguistics Research Center and is available on <http://www.clips.ua.ac.be/pages/pattern-vector#classification>

accuracy of Gossiping data dropped by 2 percent. Table 3 shows the results of the classifier.

Table 3. Results using the combined feature set

	Baseline	SVM Classifier
Gossiping	0.69	0.67
Boy_Girl	0.57	0.70
WomenTalk	0.55	0.75
Average	0.60	0.71

The numbers show that the feature set can successfully assist in the classification of texts in Boy\_Girl and WomenTalk. However, the accuracy of classification on Gossiping data perform two percent lower than baseline. There are a few possibilities to why there would be a difference between these three sets of data.

##### 1. The degree of diversity of the topics

The three boards, though all discussion oriented, involves the exchange of information in different topics. For Boy\_Girl board, most of the topic is centered on romantic relationships. As for WomenTalk board, most of the discussions focus on things that girls care about, such as products for women, boyfriends, etc. These two boards might have a clearer group of users than Gossiping, where all kinds of questions could be relevant. The topics cover from debates on international political events to opinions on superhero characters. In previous studies in English (Hasan and Ng, 2013; Hasan and Ng, 2014; Faulkner, 2014), domains are usually selected and separated so that the classification is performed on one central idea, such as gay rights or death penalty. The variety of topics might be a reason why classification on Gossiping data is less accurate than the others.

##### 2. Different language use due to the different culture of the board

Since each board on PTT has its own purpose of discussion, every board attracts different group of users and forms its unique "culture". In general, speakers on Gossiping board is more direct and more quick to criticize than users on the other two boards, as indicated by the different proportions of push comments and boo comments in the three boards. The difference might suggest that boo comments on WomenTalk and Boy\_Girl would

have a higher degree of disagreement than the ones on Gossiping, which makes it harder to differentiate push and boo comments on Gossiping. Other than possible differences among the boards, error analysis is also done by randomly selecting comments that are mistakenly tagged by the classifier. The result shows that the following mistakes are most common.

### 1. Context dependent comments

Since comments on PTT are usually left very short so that people can grasp the idea at a quick glance, a lot of words are often omitted in comments. The other users would have to judge the stance of the comment by combing the information they get from the original post and the self-tagged stance. Thus, two kinds of confusion might arise when we have to judge the stance of the comment without its context, including the original post and previous comments.

First type of error occurs when the target of the comment is not the original poster but the person or event of which the poster is attacking. For example, when a boyfriend complains about his girlfriend who always threatens to break up with him whenever they have a fight, other users might leave comments criticizing that girlfriend. But since they agree with the original poster's position, which is a negative attitude towards the girlfriend's behavior, the tags they give to their comments are usually "push". For our classifier, this would cause confusion because the linguistic behavior corresponds to negative evaluation, which is usually associated with "boo" comments. As a result, these comments would be categorized as "boo" comments. The following is an example of this type of error.

(2)	很	不	喜歡	那種	
	<i>hěn</i>	<i>bù</i>	<i>xǐhuān</i>	<i>nàzhǒng</i>	
	very	not	like	those.kind	
	婚前		就在	說	離婚
	<i>hūnqián</i>		<i>jiùzài</i>	<i>shuō</i>	<i>lihūn</i>
	before.marriage	already	say		divorce
	後	怎樣	怎樣		的
	<i>hòu</i>	<i>zěnyàngzěnyàng</i>			<i>de</i>
	after	how	DE		people

"I really don't like those people who already starts talking about what would happen when they get a divorce before even getting married"

In example (2), the comment expresses a negative attitude towards people who appears to be planning their divorce before even getting married.

Confusion may result because the target of the comment could be the person the original post was criticizing or it could be the original poster him/herself. That target could only be identified with consideration of what was originally written in the post.

The second type of error involves comments that are very short and give very little clue on their stance. The expressions that occur in these comments can be either positive or negative, depending on the speaker's intention. An example of this type of expression would be 天啊 *tiān a* "Oh my goodness", which could be used to express surprise in both positive or negative context. Since we cannot examine "how" the user says it in his/her mind and can only rely on the relative frequency of these phrases in comments, it also results in confusion.

### 2. Sarcastic comments

It is not uncommon for users to use sarcasm to express their stance online. On PTT, users might use very positive sentences and give it a negative tag to indicate that the comment was sarcastic. These negative comments might be mistaken by the classifier as "push" comments. The following example illustrates how a negative comment might be mistakenly tagged as "push" comment.

(3)	當了	鄉民	這麼	多
	<i>dāngle</i>	<i>xiāngmín</i>	<i>zhème</i>	<i>duō</i>
	be	PTT.user	this	many
	年，	我	終於	搶到
	<i>nián</i>	<i>wǒ</i>	<i>zhōngyú</i>	<i>qiǎngdào</i>
	year	I	finally	get
	頭噓	了	好	感動
	<i>tóuxū</i>	<i>le</i>	<i>hǎo</i>	<i>gǎndòng</i>
	first.booLE	so		touched

"After being a PTT user for so many years, I am finally the first one to leave a boo comment in an article! I am so touched."

The comment includes the emotion 好感動 *hǎo gǎndòng* "so touched", which appears to be a positive emotion. But human readers would be able to tell that the comment was sarcastic because of the mention of 頭噓 *tóuxū*, which is used in PTT to refer to the first boo comment in an article. Thus, this comment was tagged with "push" by the classifier.

### 3. Intentionally vague comments

On PTT, in order to avoid directly referring to a person name or avoid directly saying swear words

or negative expressions, users sometimes use characters that have similar pronunciation or similar form to replace the original characters. These would result in segmentation errors and it would be very difficult to categorize because each user might have his/her own choice of characters and there isn't an exhaustive list of such words. They may also use underlines or spaces to replace the original negative expressions when the rest of the sentence makes it clear what the word should be in that position. This omission would also make it harder to categorize the comment.

Example (4) includes the phrase 甘吟釀 *gānyínniàng*, which does not exist in Chinese vocabulary but the sounds of these words are similar to the swear words 幹拎娘 *gànlīnniáng* “you mother fucker”. The person who left this comment chose to use these words instead of the conventional characters. Other users, when reading this comment, would still be able to judge what the comment intends to express. However, the classifier might judge this new “word” to be a proper noun and this may cause some mistakes.

In example (5), the underlined part is an omission of the original word 中二 *zhōng èr*. This term is used as a negative representation for juvenile behavior and mindset common among teenagers. The word could not be identified by the classifier because the omission results in segmentation error.

- (4) 甘吟釀      的      欠  
*gānyínniàng*    *de*      *qiàn*  
*gānyínniàng*    DE      asking.for  
 噓 ~ ~ ~  
*xū*  
 boo
- (5) 圍巾 醜      原      PO  
*wéijīn chǒu*    *yuán*    *PO*  
 scarf ugly original poster  
 —              二      結案  
 —              *èr*      *jié'àn*  
 (underline)    two      case.close

The other type of vague comments are produced because of the structure of PTT comments, users sometimes try to complete other people's comments by positioning their comments at certain position. These comments would only make sense when processed in combination with the rest of the comments, also known as “floors” on PTT.

#### 4. Others

Sometimes it is very difficult to identify why the original poster would choose certain tag. This could be a result of the user's own tagging mistake, or it could also be individual differences. In example (6), the comment was tagged with “boo” while the beginning of the sentence is the word push. Both the classifier and human readers would consider this sentence to be a push comment rather than a boo comment. This could be a result of the user's own tagging error. Thus would not be considered a very important issue in the current study.

- (6) 推      投幣式 女友  
*tuī*      *tóubìshì*      *nǚyǒu*  
 push      coin-op      girlfriend  
 “I agree with coin-operated girlfriend”

To further improve the classifier, the following approaches could be taken into consideration. According to Riloff and Wiebe (2003), it is important to incorporate large amount of data because infrequent words can sometimes be strong subjective clue. Thus, it might be helpful to expand the coverage of annotated data. Context of the comments should also be taken into account. If the classifier is able to capture the relationship between the target and the comment being given, the errors caused by context dependent comments could be solved.

## 5 Conclusion

The purpose of this study is to compile lexical resources in Mandarin on arguing and stance-taking and to test the applicability of these resources in machine training on stance classification. We explored related linguistic categories on how users express their stance in online comments and established three sets of features that we believe reveals speaker's subjectivity. An experiment on classifying online comments shows that the annotated wordlist could assist in the classification by raising up to 20 percent of accuracy. In order to further improve automatic classification, an analysis on the errors of our classification task is provided. Possible linguistic issues such as identifying the targets of the comments, the overall culture on the boards discussed, sarcastic comments, and problems resulting from vague comments requires further studies.

## References

- Anand, P.; Walker, M.; Abbott, R.; Tree, J. E. F.; Bowmani, R. & Minor, M. (2011). *Cats rule and dogs drool!: Classifying stance in online debate*. Proceedings of the Workshop on Computational Approaches to Subjectivity and Sentiment Analysis, 1-9.
- Dafouz-Milne, E. (2008). The pragmatic role of textual and interpersonal metadiscourse marker in the construction and attainment of persuasion: A cross-linguistic study of newspaper discourse. *Journal of Pragmatics*, 40, 95-113.
- Faulkner, A. (2014). Automated classification of stance in student essays: An approach using stance target information and the Wikipedia link-based measure. In Proceedings of the 27th International Florida Artificial Intelligence Research Society Conference, 174-179.
- Halliday, M. A. K. (1973). *Explorations in the functions of language*. London: Edward Arnold.
- Hasan, K. S. & Ng, V. (2013). *Stance classification of ideological debates: Data, models, features, and constraints*. Proceedings of International Joint Conference on Natural Language Processing, 1348-1356.
- Hasan, K. S. & Ng, V. (2014). *Why are you taking this stance? Identifying and classifying reasons in ideological debates*. Proceedings of the Conference on Empirical Methods in Natural Language Processing, 751-762.
- Hyland, K. (1998). Persuasion and context: The pragmatics of academic metadiscourse. *Journal of Pragmatics*, 30, 437-455.
- Hyland, K. (2002). Authority and invisibility: Authorial identity in academic writing. *Journal of Pragmatics*, 34, 1091-1112.
- Hyland, K. & Tse, P. (2004). Metadiscourse in academic writing: A reappraisal. *Applied Linguistics*, 25, 156-177.
- Malouf, R., & Mullen, T. (2008). Taking Sides: User Classification for Informal Online Political Discourse. *Internet Research*, 18(2), 177-190.
- Quirk, R.; Greenbaum, S.; Leech, G. & Svartvik, J. (1985). *A comprehensive grammar of the English language*. New York: Longman.
- Reynolds, B. L. & Wang, S. (2014). An investigation of the role of article commendation and criticism in Taiwanese university students' heavy BBS usage. *Computers and Education*, 78, 210-226.
- Riloff, E. & Wiebe, J. (2003). *Learning extraction patterns for subjective expressions*. Proceedings of the Conference on Empirical Methods in Natural Language Processing, 105-112.
- Shea, D. (2006). What is PTT? Retrieved from <http://www.ptt.cc/index/html>
- Shirky, C. (2011). The political power of social media: Technology, the public sphere, and political change. *Foreign Affairs*, 90, 28-41.
- Somasundaran, S.; Namata, G.; Wiebe, J. & Getoor, L. (2009). *Supervised and unsupervised methods in employing discourse relations for improving opinion polarity classification*. Proceedings of the Conference on Empirical Methods in Natural Language Processing, 170-179.
- Somasundaran, S.; Ruppenhofer, J. & Wiebe, J. (2007). *Detecting arguing and sentiment in meetings*. Proceedings of the SIGdial Workshop on Discourse and Dialogue.
- Somasundaran, S. & Wiebe, J. (2010). *Recognizing stances in ideological on-line debates*. Proceedings of the NAACL HLT Workshop on Computational Approaches to Analysis and Generation of Emotion in Text, 116-124.
- Vande Kopple, W. J. (1985). Some exploratory discourse on metadiscourse. *College Composition and Communication*, 36, 82-93.
- Walker, M. A.; Anand, P.; Abbott, R. & Grant, R. (2012). *Stance classification using dialogic properties of persuasion*. Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, 592-596.
- Wiebe, J. (1994). Tracking point of view in narrative. *Computational Linguistics*, 20, 233-287.
- Wiebe, J.; Wilson, T.; Bruce, R.; Bell, M. & Martin, M. (2004). Learning subjective language. *Computational Linguistics*, 30, 277-308.
- Wiebe, J.; Wilson, T. & Cardie, C. (2005). Annotating expressions of opinions and emotions in language. *Language Resources and Evaluation*, 39, 165-210.
- Wilson, T. (2008). *Annotating subjective content in meetings*. Proceedings of LREC.
- Wilson, T.; Hoffmann, P.; Somasundaran, S.; Kessler, J.; Wiebe, J.; Choi, Y.; Cardie, C.; Riloff, E. & Patwardhan, S. (2005). *Opinion finder: A system for subjectivity analysis*. Proceedings of HLT/EMNLP Demonstration Abstracts, 34-35.

黃金蘭、Chung, C. K.、Hui, N.、林以正、謝亦泰、  
程威詮、Lam, B.、Bond. M., 及 Pennebaker, J. W.  
(2012)：〈中文版語文探索與字詞計算字典之  
建立〉。中華心理學刊，54，185-201。

# Learning Sentential Patterns of Various Rhetoric Moves for Assisted Academic Writing

Jim Chang<sup>1</sup>, Hsiang-Ling Hsu<sup>2</sup>, Joanne Boisson<sup>2</sup>  
Hao-Chun Peng<sup>2</sup>, Yu-Hsuan Wu<sup>2</sup>, Jason S. Chang<sup>2</sup>

<sup>1</sup>Department of Computer Science

<sup>2</sup>Department of Information System and Application

National Tsing-Hua University

Hsinchu, Taiwan R.O.C. 30013

{jim, hsiang, joanne, henry.p, shanny, jason}@nlplab.cc

## Abstract

We introduce a new method for extracting representative sentential patterns from a corpus for the purpose of assisting ESL learners in academic writing. In our approach, sentences are transformed into patterns for statistical analysis and filtering, and then are annotated with relevant rhetoric moves. The method involves annotating every sentence in a given corpus with part of speech and base phrase information, converting the sentence into formulaic patterns, and filtering salient patterns for key content words (verbs and nouns). We display the patterns in the interactive writing environment, *WriteAhead*, to prompt the user as they type away.

## 1 Introduction

The British Council estimated that roughly a billion people are learning and using English around the world (Graddol, 1997), mostly as a second language, and the number has been growing. For advanced learners in university, English for Academic Purposes (EAP) plays an important role in English Specific Purposes (ESP) study. More and more Computer Assisted Language Learning (CALL) systems have been developed to help learners in academic writing, including concordancers, grammar checkers, and essay raters. Typical CALL systems assist learners *before* and *after* the writing process by providing corpus-based reference services, or returning a grade and corrective feedback (e.g., Cambridge English *Write & Improve*).

However, researchers have shown that non-native student writers may have difficulties in compos-

ing sentences and lack knowledge at discourse level (Hinds, 1990; Swales, 1990 or Connor, 1996) in academic writing. For example, (Antony, 2003) indicated that many Japanese scientists and engineers lack sufficient knowledge of commonly used structural patterns at the discourse level.

The rhetorical organization has been considered to be one of the most effective strategies of teaching technical writing and reading. The American National Standard Institute (ANSI) recommends editors or writers to state the *purpose, methods, results, and conclusions* in the document (Weil, 1970). That is, an article usually begins with a description of background information, and is followed by the target problem, solution to the problem, evaluation of the solution, and conclusion, by analyzing annotated dictionary examples and automatically tagged sentences in a corpus. As will be described in Section 4, we used the information on collocation and syntax (ICS) for example sentences from online *Macmillan English Dictionary*, as well as in the *Citeseer x* corpus, to develop *WriteAhead*. Along the same line, the second edition of the *Macmillan English Dictionary* provides a 29-page *Improve your Writing Skills* Writing Section with instruction on how to write fluently by : *adding information, comparing and contrasting, exemplifying, expressing cause and effect, expressing personal opinions, possibility and certainty, introducing a concession or introducing topics, listing items, paraphrasing or clarifying, quoting and reporting, summarizing and concluding*.

Although there are much information (such as dictionary examples) that could help to write a paper, learners may fail to generalize from examples

The screenshot shows the WriteAhead interface with a navigation bar at the top containing 'GENERAL', 'ACADEMIC' (highlighted in red), 'OVERUSE', and 'LEARNER'. Below the navigation bar is the 'WriteAhead' title. A text box contains the user's input: 'Even though many studies have reported an increased use of computers in education, there has been very little research reported on the effectiveness of such use. AIM'. Below the text box is a navigation bar with 'less', 'more patterns', 'less', 'more examples', 'write' (highlighted in red), 'edit', 'English', '英漢', and '英和'. Below this are three suggested patterns:

- [MOVE] In this PAPER , we PROPOSE 100607**  
*In this paper , we present/propose/introduce*  
*In this paper , we describe*
- [MOVE] In this PAPER we PROPOSE 48354**  
*In this paper we present/propose/introduce*  
*In this paper we describe/discuss*
- [MOVE] The AIM of this PAPER 20725**  
*The purpose/aim/goal of this paper*  
*The aim/purpose of this study*

Figure 1: Example WriteAhead session where an user typed "pp".

and apply to their own situations. Often, there are too many examples to choose from and to adapt to match the need of the learner writers. Learners could be more effectively assisted with a tool that provides concise, relevant, genre-specific suggestions as learners type away, when writing a draft. In our research, we automatically extract rhetorical patterns to assist learners in academic writing. For example, in Figure 1, the learner has already typed a sentence describing the background and problem, and then the learner types the move tag *AIM*.

Figure 2 shows the implementation of *WriteAhead* in the Google Doc environment. With this Google Docs Add-on, the user can conveniently access the *WriteAhead* functionalities, as well as common editing functions. According to the information, *WriteAhead* displays the appropriate sentential patterns and examples for the "method" extracted from a corpus, to help the learner continue writing:

- **Our ALGORITHM be BASE on** (Our approach/method is based on),
- **We ILLUSTRATE the ALGORITHM** (We illustrate the approach/method/technique),
- **The ALGORITHM be BASE on** (The method/approach/model is base on).

In this paper, we present a prototype system, *WriteAhead*, that extracts patterns that cover extensively most semantic categories in academic writing (e.g. Teufel, 2000) from an academic corpus.

Writing suggestions are given to assist student writers. *WriteAhead* extracts these sentential patterns and examples automatically by tagging and analyzing sentences in a corpus. As will be described in Section 3, we used the *Citeseer<sup>x</sup>* corpus as our source to extract sentential patterns.

These patterns are then used at run-time in *WriteAhead* for assisted writing. *WriteAhead* takes the move tag the user types in, and then retrieves, and displays patterns related to the tag to help the user write or edit a draft (Figure 1). We present a new methodology for automatically deriving patterns. *WriteAhead* is also the first interface that suggests patterns for learners while they type.

The rest of the paper is organized as follows. The related work is reviewed in the next section. Then we present our method for automatically extracting sentential patterns and examples (Section 3). As part of our evaluation, we measured the accuracy rate of suggestions generated by *WriteAhead* using published research papers unrelated to the training data (Section 4). Section 5 reports on the experiment results and we summarize our conclusion in Section 6.

## 2 Related Work

Researchers have shown that non-native student writers may have difficulties in composing sentences

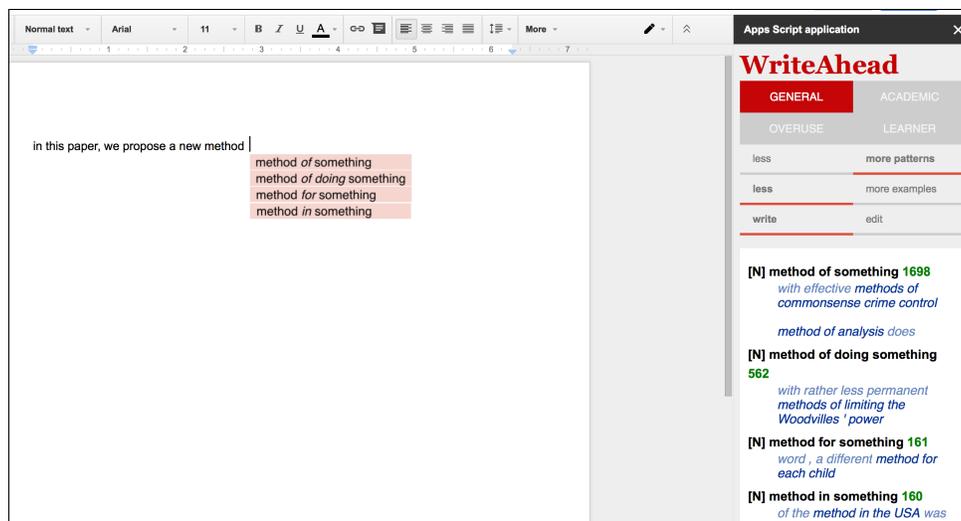


Figure 2: WriteAhead implementation for Google Doc.

and lack knowledge at discourse level (Connor, 1996 & 1999) in academic writing.

English for Academic Purposes (EAP) plays an important role in Specific Purposes (ESP) study, focusing on English of academic writing (EAW). EAW consists in numerous academic genres, including grant proposal (Connor, 1999), research articles (Swales, 1990), and reviews, which involve manual structure analysis in the academic texts for teaching academic writing. Among them, research articles (RAs) play the most significant role. In our work, we use RAs as our corpus to generate sentential patterns, which can assist learners in academic writing. We also adopt a set of semantic categories to generate patterns, which were manually identified in Teufel (1999).

## 2.1 Analysis of the Structure of Research Articles

More specifically, we focus on the structure of research articles, namely, automatically analyzing the abstracts based on series of moves. The sentences are classified to match the predefined structure. The most related body analyzing research article was Hill et al. (1982). The scheme he proposed was a starting-point for the analysis of the macrostructure of research articles. The graphical illustration of his proposed structure is like an hourglass.

Several research indicate that RAs, defined by Swales (1990), have a simpler and more clear picture

of the organizational pattern – the IMRD structure: Introduction, Method, Results and Discussion. Additionally, Swales (1981, 1990) proposed the CARS model (“Create a Research Space”) which describes the structure of the typical introductions of scientific articles according to prototypical rhetorical building plans.

The unit of analysis is the argumentative *move*, which represents “a semantic unit related to the writer’s purpose”, typically, one clause or sentence long. There is a finite number of such moves, and they are subdivided into “steps”. The model is schematically depicted in Figure 3. The model has been used extensively by discourse analysis and researchers in the field of discourse structure. Many studies adopted his theory to analyze the introduction section (e.g. Cooper, 1985; Hopkins, 1985; Crookes, 1986; Samraj, 2002, 2005). Additionally, Thompson (1993) applied it to analyze the result section while others applied it to investigate the discussion section (e.g. Hopkins and Dudley-Evans, 1994). More researches have been done to study RAs in recent years.

## 2.2 Identifying Moves for Text Classification

In the search area of automatic analysis of the discourse structure of research articles, in recent years, much work has been done viewing the task as a text classification problem that determines a label (move name) for each sentence. Various classi-

Move 1: Establishing a territory	Step 1 Claiming centrality and/or Step 2 Making topic generalization(s) and/or Step 3 Reviewing items of previous research
Move 2: Establishing a niche	Step 1A Counter-claiming or Step 1B Indicating a gap or Step 1C Question-raising or Step 1D Continuing a tradition
Move 3: Occupying the niche	Step 1A Outlining purposes or Step 1B Announcing present research Step 2 Announcing principal findings Step 3 Indicating RA structure

Figure 3: Structure of research article introduction (Swales, 1990)

fiers were applied to text categorization, including Naive Bayesian Model (NBM) (Teufel and Moens, 2002, 2004, 2006; Anthony 2003), Support Vector Machines (SVM) (McKnight and Arinivasan, 2003; Shimbo et al., 2003; Yamamoto and Takagi, 2005), Hidden Markov Model (HMM) (Lin et al., 2006), and Conditional Random Fields (CRFs) (Hirohata et al. 2008). Table 1 summarizes these approaches.

Table 1 shows the set of labels commonly used in most studies: background (B), objective (O), purpose(P), gap (G), method (M), result (R), and conclusion (C). We did not compare directly the performances of these studies, which used a different set of classification labels and evaluation data.

Anthony (2003) has developed a system which could offer a *move* analysis to assist students in writing and reading. He used the CARS model to analyze the abstracts of RA, using hand tagged RA abstracts. Shimbo et al. (2003) presented an advanced text retrieval system for MEDLINE that provides zone search specific sections in abstracts. The system classifies sentences in each Medline abstract into four sections: objective, method, results, and conclusion. Each sentence is represented by words, word bigrams, and contextual information of the sentences (e.g., class of the previous sentence, relative location of the current sentence). They reported 91.9% accuracy (per-sentence basis) and 51.2% accuracy (per-abstract basis) for the classification with the best feature set for quadratic SVM.

Similarly, Yamamoto and Takagi (2005) developed a system to classify abstract sentences into five moves, background, purpose, method, result, and conclusion. They trained a linear-SVM classifier

with features of unigram, subject-verb, verb tense, relative sentence location, and sentence score. Hirohata et al (2008) presented another system for Medline abstracts into four moves. They trained a CRF classifier with features of n-grams, sentence location, and features from previous/next sentences.

### 3 Method

Writing academic paper by referencing examples (e.g., *We illustrate the method ...*) often does not work very well, because learners may fail to generalize from examples and apply them to their own situations. Often, there are too many examples to choose from and to adapt to match the need of the learner writers. To help the learner in writing, a promising approach is to extract a set of representative sentential patterns consisting of keywords and categories that are expected to assist learners to write better.

#### 3.1 Problem Statement

We focus on the extracting process of sentential patterns for various rhetoric functions: identifying a set of candidate patterns with keywords and categories. These candidate patterns are then statistically analyzed, filtered and finally returned as the output of the system. The returned patterns can be directly examined by the learner, alternatively they can be used to annotate rhetoric moves. Thus, it is crucial that the extracted patterns cover all semantic categories of interest. At the same time, the set of extracted patterns of a semantic category cannot be too large that it overwhelms the writer or the tagging process of the subsequent move. Therefore, our goal is to return a reasonable-sized set of sentential patterns that, at the same time, must cover all rhetoric moves. We now formally state the problem that we are addressing.

*Problem Statement:* We are given a raw corpus *CORP* (e.g., *Citeseer<sup>x</sup>*) as well as an annotated corpus *TAGGED-CORP* in a specific genre and domain, and a list of semantic categories (e.g., *PAPER* = { *paper, article* }, *PRESENT* = { *present, describe, introduce* }). Our goal is to retrieve a set of tagged sentential patterns,  $p_1, \dots, p_m$ , consisting of keywords and categories from *CORP*. For this, we convert all sentences in *CORP* and *TAGGED-CORP* into candidate patterns (e.g., *In this PAPER, we PRESENT*

Table 1: Approaches of previous studies

Researchers	Model	Moves	Data
Macmillan English Dictionary	—	12 moves <sup>+</sup>	general writing
Teufel and Moens (2002, 2004, 2006)	Naive Bayes	7 moves <sup>+</sup>	scientific papers
Anthony (2003)	Naive Bayes	BPGMRC	scientific papers
McKnight and Srinivasan (2003)	Support Vector Machine	OMRC	MEDLINE
Shimbo et al. (2003)	Support Vector Machine	OMRC	MEDLINE
Yamamoto and Takagi (2005)	Support Vector Machine	BPMRC	MEDLINE
Wu et al. (2006)	Hidden Markov Model	BPMRC	Citeseer
Lin et al. (2006)	Hidden Markov Model	OMRC	MEDLINE
Hirohata et al. (2008)	Conditional Random Field	OMRC	MEDLINE

\* ADD, COMPARE, EXAM, CAUSE, OPIONION, HEDGE, TOPIC, LIST, REPHRASE, REPORT, SUM-UP  
+ AIM, TEXTUAL, OWN, BACKGROUND, CONTRAST, BASIS and OTHER

#### - Procedure ExtractPatterns(*Sent, Categories, Corpus*)

1. Extract candidate patterns from sentences in *CORP* (Section 3.2.1)
2. Group patterns by semantic categories in the given corpus (Section 3.2.2)
3. Generate sentential patterns by statistically analyzing and filtering candidate patterns (Section 3.2.3)
4. Output characteristic patterns for each category

Figure 4: Outline of the pattern extraction process

*WORK*), such that these candidates can be statistically analyzed and filtered to generate common and representative patterns.

In the rest of this section, we describe our solution to this problem. First, we define a strategy for transforming sentences from academic corpora into candidate patterns (Section 3.2.1). This strategy relies on a set of candidate patterns derived from sentences of patterns (which we will describe in detail in Section 3.2.3). In this section, we also describe our method for extracting the most representative of the candidate patterns for each semantic category of interest. Finally, we show how *WriteAhead* displays patterns at run-time (Section 3.3).

### 3.2 Transforming Sentences into Patterns

We attempt to find transformations from sentences into patterns, consisting of keywords and categories expected to characterize rhetoric moves in academic writings. Our learning process is shown in Figure 4.

#### Procedure SPs&Examples (*Sentences, Templates*)

- (1)  $taggedCorpus = ChunkParser(Sentences)$
- (2)  $candidates = GenPatternCandidate(taggedCorpus)$
- (3)  $patternInstances = ReplaceTeufel(candidates)$
- (4)  $Pats, Categories, Instances = GroupByCategory(patternInstances)$
- (5)  $Pats, Counts = Counter(Patterns, Instances)$
- (6)  $Avg, STD = CalStatics(Pats, Counts)$   
For each  $Pat, Count$  pair in  $(Pats, Counts)$   
If  $Count > Avg + MinSTDThreshold \times STD$
- (7) Emit  $Tuple = (Word, Pat, PatTuples)$
- (8)  $Pats = Annotate(Word, Pat, PatTuples)$

Figure 5: Process for extracting SPs and examples.

#### 3.2.1 Extracting Candidate Patterns.

In the first stage of the extracting process (Step (1) in Figure 5), we tokenize sentences in the given corpus, and assign to each word its syntactic information including lemma, part of speech, and phrase group (represented using the B-I-O notation to mark the beginning, inside, and outside of some phrase group).

See Table 2 for an example of tagged sentence. In order to identify the head of a phrase, we convert the B-I-O notation to I-H-O notation with *H* denoting the headword of a phrase. Using the I-H-O notation allows us to directly identify the headword of a phrase chunk. Then, we convert every word in a sentence into elements of a candidate pattern. The

Table 2: A tagged sentence, pattern elements for each word, and anchored pattern candidates

Word	Lemma	POS	B-I-O	I-H-O	Element	Pattern candidate anchored at each word
In	In	IN	B-PP	I-PP	in	(In)
this	this	DT	B-NP	H-NP	this	(In this)
report	report	NN	I-NP	H-NP	PAPER	(In this PAPER)
,	,	,	O	O	,	(In this PAPER ,)
we	we	PRP	B-NP	I-NP	we	(In this PAPER , we)
propose	propose	VBP	B-VP	H-VP	PRESENT	(In this PAPER , we PRESENT)
a	a	DT	B-NP	I-NP	(.)	(In this PAPER , we PRESENT .)
method	method	NN	I-NP	H-NP	WORK	(In this PAPER , we PRESENT _ WORK)

Table 3: Example Teufel category of sentential patterns

Teufel category	Gloss and examples
PAPER	article, draft, paper, project, report, ...
WORK	analysis, approach, method, ...
PURPOSE	aim, goal, purpose, task, theme, topic, ...
DEVELOP	accomplish, achieve, answer, prove, ...
PRESENT	describe, introduce, present, propose, ...
EVALUATE	compare, compete, evaluate, test, ...

elements consist of three types of information:

- **Semantic categories** (See Table 3) : typical domain specific concepts and words,
- **Lexical symbols**: a list of common prepositions, pronouns, adverbs, and determinants,
- **Noun phrase** and **verb phrase**: head words that are not classified in a category are represented as *something* or *do*.

Note that determinants (e.g., the, an, a) or adjectives need to be represented in a pattern. For those words, we add ”\_” (*ignored*) to the element list. The *ignored* elements will be deleted before patterns are analyzed and filtered (as shown in Table 2). We design the scope of extracted patterns, as from the beginning of the sentence, to the object phrase after the main verb.

Finally, we combine elements for a sentence to generate pattern candidates (Step (2) in Figure 5). Table 2 shows those elements associated with words and how they combine to form pattern candidates.

In Step (3), we use semantic categories to generalize words and generate formulaic patterns. As will be described in Section 4, we used a Teufel manually analyzed research article to devise a set of categories of words (Teufel, 1999).

For example, in Table 2, the sentence ”*In this paper, we propose a method that accurately reports timing information by accounting for intrusion introduced by monitoring.*” will be transformed into the candidate pattern ”*In this PAPER, we PRESENT WORK*”.

The input to this state is a set of sentences. These sentences constitute the data for generating the candidate patterns, that can be used in the next step.

The output of this stage is a set of candidate patterns that can be statistically analyzed and filter in a later step. See Table 4 for example candidate patterns extracted from some sentences.

### 3.2.2 Grouping Patterns by Categories.

In the second stage of the process (Step (4) in Figure 5), we filter candidate patterns to generate representative patterns. Once patterns and instances are generated, they are sorted and grouped by category.

Then, we count the number of instances of each pattern within the category (in Step (5)), and the average and standard deviation of these counts for each category (in Step (6)).

In Step (7), we select patterns with an instance count exceeding the average count by *MinSTDThreshold* standard deviation.

Consider the partial sentence ”*In this paper, we propose a method*” Table 2 shows elements of each word, pattern candidates anchored each word. Note that the candidate (e.g., **In this PAPER, we PRESENT WORK** associated with the instance of *In this paper, we propose a method*) are valid patterns.

Table 4: Ranked patterns based on *CORP* and *TAGGED-CORP* statistics

AZ	Pattern	Count	Example
AIM	In this PAPER , we PROPOSE	100,607	In this paper , we present/propose/introduce In this paper , we describe
AIM	In this PAPER we PROPOSE	48,354	In this paper we present/propose/introduce In this paper we describe/discuss
AIM	The AIM of this PAPER	20,725	The purpose/aim/goal of this paper The aim/purpose of this study
AIM	In this PAPER , we INVESTIGATE	16,872	In this paper , we investigate/examine/identify In this paper , we analyze

### 3.2.3 Ranking and Annotating Patterns.

In the third and final stage (Step (8) in Figure 5), we count, sort, and filter patterns, essentially using the frequency counts from *CORP* with the tags in *TAGGED-CORP* (See Tables 4). Figure 5 shows the algorithm for ranking a set of corresponding sentential patterns for all semantic categories. See Table 6 for an example of the move tag *AIM* and its corresponding sentential patterns.

### 3.3 Run-Time

Once the patterns and examples are automatically extracted for each category in the given corpus, they are stored and indexed by category that can be annotated with corresponding rhetoric moves. At run-time in a writing session, *WriteAhead* detects a rhetoric move tag *Move* in the text box. With the tag as a query, *WriteAhead* retrieves and sorts all relevant patterns and examples (*Pattern* and *Example*) by frequency, aiming to display the most common information toward the top.

## 4 Implementation and Setting

In this section, we describe the implementation and experiments of the method presented in Section 3. First, we retrieved computer science abstracts from the digital library website, *CiteSeer* ([citeseerx.ist.psu.edu](http://citeseerx.ist.psu.edu), a collection of bibliographies of scientific literature in computer science from various sources). We obtained about four million computer science abstracts. Before extracting patterns, we use GeniaTagger as a simple toolkit for analyzing and parsing English sentences and outputting the base forms, part-of-speech tags, chunk tags and named

entity tags. Tsuruoka et al. (2005) develop this tools, specifically tuned for biomedical text. Then, we use this tagger for tagging sentences to obtain the part-of-speech tags, lemma, chunks for training the model. After tagging, we applied our method to extract patterns from *CiteSeer x*.

We used manually compiled semantic categories and words (Teufel, 1999) to generalize word and generate formulaic patterns. There are 66 categories with some 900 nouns, verbs, and adjectives.

For example, in Table 2, the sentence "In this paper, we propose a method that accurately reports timing information by accounting for intrusion introduced by monitoring." will be transformed into the candidate pattern "In this PAPER, we PRESENT WORK". As will be described in Section 4, we used a Teufel manually analyzed research article to derive a set of categories of words (Teufel, 1999).

We also used the Argumentative Zoning corpus (available at [www.cl.cam.ac.uk/~sht25/AZ\\_corpus.html](http://www.cl.cam.ac.uk/~sht25/AZ_corpus.html)), created and annotated by Teufel (2010) and collaborators. The dataset consists of 80 AZ-annotated conference articles in the research area of computational linguistics, hosted in the academic archive of Cmplg arXiv. The set of AZ tags include AIM (purpose), BAS (basis), BKG (background), CTR (contrast), OTH (previous work), OWN (own method and results), TXT (textual references).

To provide user interface for accessing *WriteAhead*, we have implemented two versions of the system: (1) browser-based proof-of-concept (POC) interface and a Google Docs add-on. (2) To prove the feasibility of the concept, the browser version (with-

out the common editing functions) was implemented in Python and within the Flask Web framework. We stored the suggestions in JSON format using PostgreSQL for a faster access. The *WriteAhead* server obtains client input from a popular browser (Safari, Chrome, or Firefox) dynamically with AJAX techniques. For uninterrupted service and ease of scaling up, we chose to host *WriteAhead* on Heroku, a cloud-platform-as-a-service (PaaS) site.

The *WriteAhead* add-on for Google Docs was implemented in *Google App Script (GAS) with HTML and JavaScript*. *WriteAhead* add-on obtains client input from documents using built-in methods in GAS, and obtains the suggestions by sending requests to the *WriteAhead* server through our API. To start the add-on, the user click "Add-ons > WriteAhead > Start" after installation, and a sidebar will appear. The suggested patterns and examples are shown in the sidebar.

## 5 Evaluation

The proposed system was designed to automatically extract patterns and examples for each corresponding rhetoric move. A preliminary evaluation was done on a set of real formula labeled with moves. We compared our patterns with the Teufel formula and evaluated the precision in different experimental settings. In this Section, we first describe how we compared patterns (Section 4). Then, Section 5.1 introduces the evaluation metrics for evaluating the experimental results.

### 5.1 Evaluation Metrics

The output of our method is an automatically tagged pattern, which can either be shown to the user directly, or be used in academic writing, e.g., in teaching academic writing, teachers can use those tagged patterns to help tag the sentences in a given RA with moves.

To evaluate our approach, we compare our sentential patterns with Teufel Formula. We extract three categories PAPER, WORK, and PRESENT to inspect that if there are some patterns in common.

We compare Teufel formulas and our extracted sentential patterns under three specific categories, PAPER, WORK, and PRESENT. In PAPER category, there are 76 formula and 5 patterns, among

these, 18 formula and 5 patterns are in common. Similar in WORK category, there are 153 formula and 16 patterns, among these, 30 formula and 16 patterns are in common. And in PRESENT category, there are 15 formula and 13 patterns, among these, 7 formula and 5 patterns are in common.

In comparison, our sentential patterns are longer and more complete than Teufel's original formulas serving as features for classification. It is clear that, our sentential patterns tend to be more complete and indicative of rhetoric moves, since we use the stopping condition to extract patterns all the way up to the object of the main verb. As it turns out, the patterns generate by a computer using a very large scale academic corpus are more consistent, complete, and relevant to academic writing.

## 6 Conclusion and Future Work

Many avenues exist for future research and improvement of our system. For example, the method for extracting patterns could be discussed further and be evaluated separately, using different formula to calculate a threshold, and generate different patterns. Additionally, an interesting direction to explore is to expand the word categories and obtain more fine-grain patterns. Yet another direction of this research would be applied to the model of different sections of RAs, and more disciplines.

In summary, we have presented a new method for extracting patterns in a scholar big dataset for various moves in academic writing. The method involves patterns with all semantic categories of interest. The experimental results show that our automatically extracted patterns reflect different rhetoric moves and purposes.

## References

- Anthony, Laurence and Lashkia, George V. 2003. Mover: A machine learning tool to assist in the reading and writing of technical papers. *Transactions on Professional Communication, IEEE*. Vol 46,3, pages 185–193, IEEE.
- Connor, Ulla. 1996. Contrastive rhetoric: Cross-cultural aspects of second language writing. *Cambridge University Press*.
- Connor, Ulla and Mauranen, Anna. 1999. Linguistic analysis of grant proposals: European Union research

- grants. *English for specific purposes*. Vol. 18, 1, pages 47–62, Elsevier.
- Cooper, Catherine. 1985. Aspects of article introductions in IEEE publications. *Unpublished master's thesis, University of Aston, Birmingham, England*.
- Crookes, Graham. 1986. Towards a validated analysis of scientific text structure. *Applied linguistics, An Association of Applied Linguistics*. Vol. 7, 1, pages 57–70.
- Dudley-Evans, Tony. 1994. Genre analysis: An approach to text analysis for ESP. *Advances in written text analysis*. Vol. 219, page 228.
- Graddol, David. 1997. The future of English?: A guide to forecasting the popularity of the English language in the 21st century. *British Council*.
- Hill, Susan S and Soppelsa, Betty F and West, Gregory K. 1982. Teaching ESL Students to Read and Write Experimental-Research Papers. *TESOL quarterly*. Vol. 16, 3, pages 333–347, Wiley Online Library.
- Hinds, John. 1990. Inductive, deductive, quasi-inductive: Expository writing in Japanese, Korean, Chinese, and Thai. *Coherence in writing: Research and pedagogical perspectives*, pages 87–110.
- Hirohata, Kenji and Okazaki, Naoaki and Ananiadou, Sophia and Ishizuka, Mitsuru and Biocentre, Manchester Interdisciplinary. 2008. Identifying Sections in Scientific Abstracts using Conditional Random Fields. *IJCNLP*, pages 381–388.
- Hopkins, Andy. 1985. An investigation into the organizing and organizational features of published conference papers. *Unpublished MA dissertation, University of Birmingham*.
- opkins, Andy and Dudley-Evans, Tony. 1988. A genre-based investigation of the discussion sections in articles and dissertations. *English for Specific Purposes*. Vol. 7, 2, pages 113–121, Elsevier.
- Lin, Jimmy and Karakos, Damianos and Demner-Fushman, Dina and Khudanpur, Sanjeev. 2006. lin2006generative. *Association for Computational Linguistics*, pages 65–72.
- McKnight, Larry and Srinivasan, Padmini. 2003. Categorization of sentence types in medical abstracts. *American Medical Informatics Association Annual Symposium Proceedings 2003*, page 440.
- Ruiying, Yang and Allison, Desmond. 2003. Research articles in applied linguistics: Moving from results to conclusions. *English for Specific Purposes*. Vol. 22, 4, pages 365–385.
- Samraj, Betty. 2002. Introductions in research articles: Variations across disciplines. *English for specific purposes*. Vol. 21, 1, pages 1–17, 2002. Elsevier.
- Shimbo, Masashi and Yamasaki, Takahiro and Matsumoto, Yuji. 2003. Using sectioning information for text retrieval: a case study with the medline abstracts. *Proceedings of Second International Workshop on Active Mining (AM'03)*.
- Swales, John. 1990. Genre analysis: English in academic and research settings. *Cambridge University Press*.
- Teufel, S., Carletta, J. & Moens, M. 1999. An annotation scheme for discourse-level argumentation in research articles. *EACL*.
- Teufel, Simone. 2000. Argumentative zoning: Information extraction from scientific text. *Diss. University of Edinburgh*.
- Teufel, Simone and Moens, Marc. 2002. Summarizing scientific articles: experiments with relevance and rhetorical status. *Computational linguistics*. Vol.28, pages 409–445. MIT Press.
- Teufel, Simone. 2010. The Structure of Scientific Articles: Applications to Citation Indexing and Summarization (Center for the Study of Language and Information-Lecture Notes). *Center for the Study of Language and Inf.*
- Thompson, Dorothea K. 1993. Arguing for Experimental Facts in Science A Study of Research Article Results Sections in Biochemistry. *Written communication*. Vol. 10, 1, pages 106–128, Sage Publications.
- Tsuruoka, Yoshimasa and Tateishi, Yuka and Kim, Jindong and Ohta, Tomoko and McNaught, John and Ananiadou, Sophia and Tsujii, Junichi. 2005. Developing a robust part-of-speech tagger for biomedical text. *Advances in informatics*, pages 382–392, Springer.
- Weil, Ben H. 1970. Standards for writing abstracts. *Journal of the American Society for Information Science*. Vol. 1, 5, pages 351–357, Wiley Online Library.
- Yamamoto, Yasunori and Takagi, Toshihisa. 2005. A sentence classification system for multi biomedical literature summarization. *Data Engineering Workshops, 2005. 21st International Conference on Biomedical Data Engineering*, pages 1163–1163, IEEE.

## Idioms: Formally Flexible but Semantically Non-transparent

**Hee-Rahk Chae**

Hankuk Univ. of Foreign Studies  
Oedae-ro 81, Mohyeon, Yongin  
Gyeonggi 17035, Korea  
hrchae@hufs.ac.kr

### Abstract

Contrary to popular beliefs, idioms show a high degree of formal flexibility, ranging from word-like idioms to those which are like almost regular phrases. However, we argue that their meanings are not transparent, i.e. they are non-compositional, regardless of their syntactic flexibility. In this paper, firstly, we will introduce a framework to represent their syntactic flexibility, which is developed in Chae (2014), and will observe some consequences of the framework on the lexicon and the set of rules. Secondly, there seem to be some phenomena which can only be handled under the assumption that the component parts of idioms have their own separate meanings. However, we will show that all the phenomena, focusing on the behavior of idiom-internal adjectives, can be accounted for effectively without assuming separate meanings of parts, which confirms the non-transparency of idioms.

### 1 Introduction

Although idioms are generally assumed to be non-compositional and, hence, non-flexible, it has been well attested that they are not fixed expressions formally. Even one of the most fixed idioms like [*kick the bucket*] show morphological flexibility in the behavior of the verb *kick*. Many other idioms show some degree of syntactic flexibility with reference to various types of syntactic behavior. Even the non-compositionality of them has been challenged, especially by those who are working under the framework of cognitive linguistics (cf.

Croft & Cruse 2004: Ch. 9, and Gibbs 2007). Reflecting this trend, Wasow et al. (1983) and Nunberg et al. (1994), for example, argue that syntactic flexibility is closely related to semantic transparency. In this paper, however, we are going to show that idioms can better be analyzed as semantically non-transparent although they are formally flexible, providing further evidence for the analysis in Chae (2014).

Adopting Culicover's (2009) definition of construction,<sup>1</sup> Chae (2014) assumes that all and only idioms are represented as constructions. Under this view, grammar consists of three components: the set of lexical items (i.e. the lexicon), the set of rules and the set of constructions. He introduces some "notations/conventions," which apply to regular phrase structures, to represent the restrictions operating on idioms. Employing these notations, he provides representations of various types of formal properties of idioms (in English and Korean): from the least flexible ones to the most flexible ones. However, the meanings of idioms are supposed to come from the whole idioms/constructions rather than from their component parts compositionally.

In section 2, we will introduce a framework to represent the syntactic flexibility of idioms, which is developed in Chae (2014). We will also observe some consequences of the framework on the lexicon and the set of rules. Then, in section 3, we will examine some phenomena which seem to be handled only by assuming that the component parts of idioms have their own separate meanings. It will be shown, however, that all the phenomena can be accounted for effectively without assuming

---

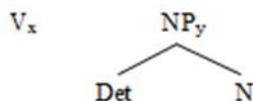
<sup>1</sup> The definition is as follows (Culicover 2009: 33): "A construction is a syntactically complex expression whose meaning is not entirely predictable from the meanings of its parts and the way they are combined in the structure."

separate meanings of parts. We will focus on the behavior of idiom-internal adjectives, which is the most difficult to treat properly under the assumption of semantic non-transparency of idioms.

## 2 Formal Flexibility

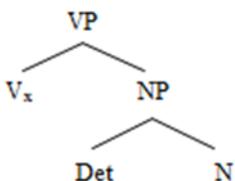
Traditionally idioms are classified into two classes: “decomposable idioms”/ “idiomatically combining expressions (ICEs)” and “non-decomposable idioms”/ “idiomatic phrases (IPs)” (Nunberg 1978, Nunberg et al. 1994, Jackendoff 1997, Sag et al. 2002, etc.). Jackendoff (1997: 168-169) analyzes the two classes as follows:

(1) A decomposable idiom: *bury the hatchet*



[RECONCILE ( [ ]<sub>A</sub>, [DISAGREEMENT]<sub>y</sub>)]<sub>x</sub>

(2) A non-decomposable idiom: *kick the bucket*



[DIE ( [ ]<sub>A</sub>)]<sub>x</sub>

In the former, which has the meaning of ‘reconcile a disagreement’ or ‘settle a conflict,’ the two component parts *bury* and [*the hatchet*] are assumed to have their own meanings and are separated from each other syntactically because the NP can be “moved” around. In the latter, no component parts have separate meanings and they are all connected syntactically.

Espinal & Mateu (2010: 1397), however, argue that the distinction is “not as clear-cut and uniform as has been assumed.”

- (3) a. i) John laughed his head off.  
ii) We laughed our heads off.
- b. Bill cried his eyes out on Wednesday, and he cried them out again on Sunday.
- c. i) \*Whose/which heart did Bill eat out?  
ii) \*His heart, Bill ate out.

- d. i) \*Bill ate his [own/inner heart] out.  
ii) \*We were laughing our [two heads] off.

The examples in (a) and (b) show ICE-like properties. On the other hand, those in (c) and (d) show their IP-like properties. In addition, Wulff (2013: 279) makes it clear that idioms are not to be classified into separate categories: “..., resulting in a ‘multi-dimensional continuum’ of differently formally and semantically irregular and cognitively entrenched expressions that ultimately blurs the boundaries of idiom types as described in Fillmore et al. (1988) and various other, nonconstructionist idiom typologies ...”

According to Chae (2014: 495-6), however, Espinal & Mateu’s (2010) analysis is not very reasonable, either. They argue all the internal elements of idioms have metaphoric/non-literal meanings and the meanings of the whole idioms can be derived from them compositionally. First of all, it is not clear how the metaphoric meanings of the internal elements can be obtained. Hence, we will need a framework which is formal enough to be computationally useful, and which is flexible enough to handle all the (morphological and syntactic) idiosyncrasies of idioms. For this purpose, Chae (2014) provides a system for the representation of idiomatic constructions.<sup>2</sup>

Based on the fact that idiomatic expressions are typical examples showing irregularities on various levels, Chae (2014: 501) introduces four notations/conventions to indicate lexical and formal restrictions operating on idioms:

- (4) a. <...>: the phrase is a syntactic “island” (no extraction is allowed).
- b. /.../: the phrase cannot be further expandable by internal elements.
- c. {...}: only the lexical items listed inside the brackets are allowed to occur.
- d. CAPITALIZATION: lexical items in capital letters have to be inflected for their specific forms.

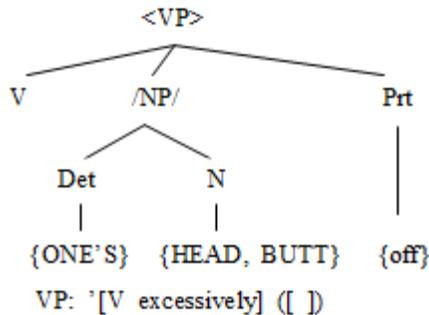
The former two are used to restrict (external and internal) syntactic behavior, and the latter two to regulate lexical and morphological behavior.

<sup>2</sup> The system was developed on the basis of English idioms. However, its main purpose was to analyze Korean data in such idiom dictionaries as No (2002) and Choi (2014). The system has been proved to be very successful in representing Korean idioms and, hence, would be effective in analyzing idioms in other languages as well.

Employing the notations in (4), Chae (2014) provides analyses of various types of idioms (in English and Korean) on the basis of their formal properties: from the least flexible ones to the most flexible ones. Please note that the notations apply to regular phrase structures. Regular properties of idioms are captured by way of these phrase structures and their irregular properties are captured with reference to the notations.

We can analyze the [V *one's head off*] idiom in (3) as follows, under our representational system.

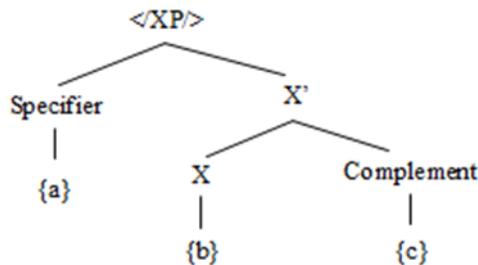
(5) A new analysis of [V *one's head off*]



The <...> on the VP indicates that no internal elements can be extracted out of the VP. The /.../ on the NP indicates that the node cannot be expanded further. The {...} under lexical categories indicates that only those lexical items inside it are allowed in the position. Under the N node, two lexical items are listed, which means that any of them is allowed in that position. The position under V is open because it has no {...}. As the lexical items under the Det and N are capitalized, they are required to have specific inflectional forms in actual sentences.

Under the present framework, one of the most rigid idioms can be represented as follows (Chae 2014: 505-6):

(6)



As the XP has both <...> and /.../, no elements inside it can be extracted outward, and it cannot be further expanded internally. In addition, all the

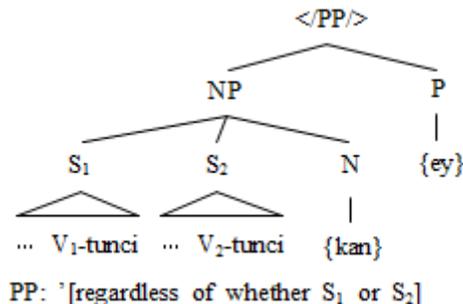
lexical items are enclosed with the notation {...}, which has only a single member. The flexibility will increase as more lexical items are capitalized, as more lexical items appear in {...}, and as <...> or /.../ disappears, eventually to become regular/non-idiomatic phrases.

We assume that the framework introduced in Chae (2014) is formal enough to be computationally useful and flexible enough to handle various types of formal properties of idioms. The behavior of idioms is regular to the extent that they are represented on phrase structures, and is irregular to the extent that their structures are regulated by the notations in (4).

The present framework has the effect of simplifying the main components of the grammar of a language, namely, the lexicon and the rule set. Firstly, the framework makes it possible to reduce the number of lexical items or their senses. For example, [*ei(-ka) eps-* 'be preposterous'] in Korean is a typical idiom. Its literal meaning is something like 'there is no EI.' The "word" *ei*, although it can be followed by the nominative marker *-ka*, does not have its own meaning and it can only be used as a part of the idiom. Korean dictionaries list both the idiom [*ei(-ka) eps-*] and the word *ei* as separate entries, which is necessary because an adverb like *cengmal* 'really' can be inserted between *ei(-ka)* and *eps-*, [*ei(-ka) cengmal eps-* 'be really preposterous']. Under the present approach, however, *ei* does not have to be listed as a separate entry and, hence, we can reduce the number of lexical items. We do not need the putative word because the construction representing the idiom is flexible enough to allow regular adverbs in between the two parts of the idiom, as we can see in chapter 3.

Next, let us consider how we can simplify the rule set. The expression [... *V<sub>1</sub>-tunci* ... *V<sub>2</sub>-tunci kan-ey* 'regardless of whether S<sub>1</sub> or S<sub>2</sub>'] is an idiom (No 2002: 268). It can be analyzed as follows:

(7)



The idiom has not only semantic anomalies but also syntactic anomalies. No other NPs in Korean have the structure of  $[S_1 S_2 N]$ , in which the head noun has two sentential complements. If we are going to handle the structure with phrase structure rules, we have to posit a rule of the following:  $[NP \rightarrow S S N]$ . This NP is very special in the sense that it has a unique internal structure of  $[S_1 S_2 N]$ . In addition, as it can occur only before the postposition *-ey*, its distribution is severely limited. These facts will render the rule set very complex. Under our approach, however, the construction in (7) is listed in the set of constructions. Then, we do not have to account for the special properties of the idiom with complex phrase structure rules and unmotivated stipulations.

### 3 Semantic Non-transparency

We assume that idioms are syntactically flexible and semantically non-transparent. We have seen that their syntactic flexibility can be handled effectively with the framework introduced in Chae (2014). In this section, we will focus on their semantic non-transparency. This assumption is based on the observation that the component parts of idioms do not have independent meanings and, hence, that the meaning of the whole idiom cannot be obtained from their parts compositionally. We will first review the issue over the compositionality of idiomatic expressions. Then, we will observe those phenomena which have led some scholars to assume that the component parts have separate meanings. Finally, we will develop a framework in which we can handle the phenomena, especially idiom-internal modification, without assuming separate meanings of parts.

#### 3.1 The Issue: Compositionality

There has been a controversy over the issue whether idioms conform to the principle of compositionality or not. From a non-compositional point of view, for example, Nicolas (1995) argues that the parts of an idiom do not have individual meanings. Schenk (1995) argues that there is no relation between the meaning of the whole idiom and the meanings of its parts. In addition, Culicover & Jackendoff (2005: 34) makes it clear that “there is no way to predict the meanings of ... from the words” in various types of “lexical VP idioms” (cf. Goldberg 1995). However, many other works stand on the other side: Wasow et al. (1983), Nunberg et al. (1994), Geeraerts (1995), Gibbs (1995), Sag et al. (2002), Espinal & Mateu (2010), and others.

One of the most difficult challenges of compositional approaches lies in figuring out the meanings of the parts of an idiom. For example, it is generally assumed that the meanings of *spill* and *beans* in the idiom [*spill the beans*] are ‘divulge’ and ‘information,’ respectively. However, it is unlikely that we can get at their meanings, if there are individual meanings, without consulting the meaning of the whole idiom (cf. Geeraerts 1995, Gibbs 2007: 707). Then, we do not have to worry about the meanings of individual parts from the beginning, because the reason we need to know the individual meanings is to compute the meaning of the whole idiomatic meaning. Although there are many cognitive linguistic approaches which seek to obtain the meanings of the parts on the basis of people’s conceptual knowledge, they can only provide partial answers, as is hinted in Gibbs (2007: 709, 717). From a computational point of view, partial answers would be largely the same as no answers.

Non-compositional approaches may run into difficulties as well, in such cases as the following: i) when a part of the idiom is displaced from its “original” position (cf. (8d)), and ii) when a part is modified (cf. (9-10)). In these cases, it would be very difficult to compute the idiomatic meaning without recourse to the meanings of the individual parts, especially under surface-oriented frameworks (cf. Wasow et al. 1983). Without handling these cases appropriately, a non-compositional approach would not be viable.

When an internal element of an idiom is displaced from its original position, as in [*the hatchet we want to bury after years of fighting*], it would not be easy to capture the idiomatic meaning in surface-oriented frameworks, which do not have “underlying” structures, because parts of the idiom are separated from each other by a syntactic operation. Note that idioms are generally assumed to be word-like fixed expressions in previous non-compositional approaches. However, in our approach, the identity of the idiom can be captured with reference to the construction describing the idiom, which is in the set of constructions.

As for idiom-internal modifiers, there are three types to be considered. Firstly, adverbs can occur before verbs inside some idioms. Secondly, adjectives can occur before nouns in a few idioms and function as nominal modifiers. Thirdly, adjectives in some idioms function, surprisingly, as verbal modifiers. More surprisingly, Nicolas (1995) shows that most idiom-internal adjectives function as verbal modifiers, i.e. they have the

function of modifying the whole idiom or its predicate. We have reached largely the same conclusion after examining the idioms in two Korean idiom dictionaries: No (2002) and Choi (2014).

Although English does not seem to have examples of the first type, Korean has some. This difference may be due to the word order difference between the two languages: Korean is a head-final language, while English is a head-initial language. As an adverb occurs before the string of V-NP in English, it is not clear whether it modifies V or VP. In addition, regardless of whether it modifies V or VP, the effect is the same. Even when it modifies V, the influence will go over to the whole VP because V is the head of VP. On the other hand, an adverb can occur inside the NP-V string in Korean, which clearly shows that it modifies V. As we saw above, the Korean idiom [*ei(-ka) eps-* ‘be preposterous’] can be modified by an internal adverb such as *cengmal* ‘really,’ [*ei(-ka) cengmal eps-* ‘be really preposterous’] (cf. Chae 2014: 511). When the internal modifiers are adverbs, it is not very surprising that they have the function of modifying the whole idiom, because the modified element, i.e. V, is the head of VP.

### 3.2 Any Compositional Phenomena?

To begin with, we want to make it clear that we cannot derive the meanings of idioms from their component parts compositionally. It is a well-known fact that we can guess the meanings of component parts only when we know the meaning of the whole idiom (cf. Gibbs 2007: 709, 717). For example, we cannot usually figure out that the meanings of *spill* and *beans* in the idiom [*spill the beans*] are ‘divulge’ and ‘information,’ respectively, unless we know the meaning of the whole idiom, i.e. ‘divulge information.’ If it is not the case, those who are learning English would predict the meaning of the idiom correctly on the basis of the (literal) meanings of *spill* and *beans*, which is very unlikely. If we can only figure out the individual meanings with reference to the meaning of the whole idiom, we do not have to worry about the meanings of individual parts from the beginning. As we all know, we need to know the meanings of individual words to compute the meaning of the whole expression.

Despite the problems described above, there has been a tradition which takes it for granted that individual words in idioms have to have their own meanings. Nunberg et al. (1994: 500-3) is one of the forerunners: “modification, quantification, topicalization, ellipsis, and anaphora provide

powerful evidence that the pieces of many idioms have identifiable meanings which interact semantically with other” (cf. Wasow et al. 1983; Croft & Cruse 2004: ch. 9, Gibbs 2007).

- (8) a. [kick the filthy habit]  
 b. Pat got the job by [pulling strings that weren’t available to anyone else].  
 c. [touch a couple of nerves]  
 d. Those strings, he wouldn’t pull for you.  
 e. My goose is cooked, but yours isn’t \_\_\_\_.  
 f. Although the FBI kept taps on Jane Fonda, the CIA kept them on Vanesa Redgrave.

It would be very difficult to account for these data if we do not assume that individual words in idioms have their own meanings. In (a-b), at least formally, a part of the idiom is modified. In (c), a part is quantified. In (d), a part is topicalized. In (e-f), an anaphor or a deleted part refers to a part of the idiom concerned.

Under traditional approaches, we would not be able to account for the phenomena in (8) appropriately unless we assume that individual words have their own identities. Under the spirit of Chae (2014), however, we can account for the phenomena in (c-f) easily. We are assuming that all and only idioms are represented as constructions and that constructions can represent formal flexibilities of idioms. In our analysis of the idiom in (c), the position of Det/QP is open in the construction concerned.<sup>3</sup> For the constructions in (d-f), the syntactic mechanisms involved, i.e. those responsible for figuring out the antecedents of gaps or anaphora, will identify the relevant entities. For example, [*those strings*] will be identified as the object NP of *pull* in (d) and *yours* will be identified as *your goose* in (e). Then, the idiom concerned will be identified with reference to the construction describing it, which is in the set of constructions. That is, the relevant construction will be invoked and, hence, its meaning as well, without recourse to the individual words involved.

To be more specific about the topicalized example in (8d), it can be analyzed the same way as other topicalized sentences. Just as a regular VP which has a displaced object is analyzed as VP/NP, the idiomatic VP [*pull e*], which has its own idiomatic meaning and is lacking [*those strings*], is analyzed as VP/NP. When the missing NP, i.e. the

<sup>3</sup> If different determiners and/or quantifiers allowed in the idiom result in different meanings, such data could be handled with the mechanisms for idiom-internal adjectives in section 3.3.

NP value of the SLASH(/) feature, gets licensed, the whole idiom obtains its meaning from the construction concerned. This is not possible in previous non-compositional approaches because idioms are generally assumed to be word-like fixed expressions.

The difficulty lies in the analysis of such data as those in (8a-b). As an adjective or a relative clause modifies a noun which is a part of the idiom, there does not seem to be an easy way of accounting for the data without assuming that all the component parts of the idiom have their meanings. However, in the next section, we will see that the data can better be analyzed without such an assumption.

### 3.3 Idiom-internal Modification

Among the three types of idiom-internal modifiers mentioned in section 3.1, we will consider how we can account for the second and third types, i.e. the behavior of idiom-internal adjectives. We will see that the framework to be developed here can handle the phenomena without assuming separate meanings of idiom parts. This implies that idiom-internal modifiers are not part of the idiom. It will be also shown that the meaning of the whole expression can be obtained compositionally from that of the idiom and that of the modifier.

With reference to such data as in (8a-b), Nicolas (1995: 233, 239-10) argues that the internal modification in V-NP idioms “is systematically interpretable as modification of the whole idiom.”<sup>4</sup>

- (9) a. [make rapid headway] ‘progress rapidly’  
 b. [be at a temporary loose end]  
 ‘be unoccupied temporarily’  
 c. [pull no strings] ‘do not exert influence’

In these examples all the idiom-internal adjectives are interpreted as adverbials. It seems to be true that most of the adjectives in idioms have the function of modifying the whole idiom.

However, there are some examples where the idiom-internal adjective does not have an adverbial function, including those in (8a-b).

- (10) a. [bury the old/bloody/violent hatchet]  
 ‘settle an old/bloody/violent conflict’  
 b. [bury the ancestral hatchet]  
 ‘reconcile an ancestral disagreement’  
 c. [spill the salacious beans]  
 ‘divulge the salacious information’

<sup>4</sup> Nicolas (1995: 244, 249) even argues that he could not find any counter-examples to the adverbial function of adjectives in his chosen corpus of fifty million words.

In all these examples, the underlined adjectives have the function of modifying some nominal elements of the meanings of the whole idiom. Please note that they do not have the following meanings:

- (11) a. ‘settle a conflict  
in a(n) old/bloody/violent way’  
 b. ‘ancestrally reconcile a disagreement’  
 c. ‘salaciously divulge information’

In the case of the idiom [*bury the hatchet*], the adjective *official* leads to an adverbial function: [*bury the official hatchet*] ‘settle/reconcile a conflict/disagreement officially.’ On the other hand, as we can see in (10a-b), the adjectives *old/bloody/violent/ancestral* induce an adjectival function in the idiom. This shows that the function of an idiom-internal adjective is determined by the interactions between the adjective and the idiom within which the adjective is located. The issue, then, is how we can account for the adverbial and adjectival functions of idiom-internal adjectives without assuming that the component parts of an idiom have separate meanings.

As the first step to the solution, let us examine the characteristics of the idiom [*bury the hatchet*] more closely. When it contains an adjective inside, the adjective can be interpreted either as adjectively or as adverbially. As we can see in (10a-b), such adjectives as *old*, *bloody*, *violent* and *ancestral* lead to an adjectival reading. In [*bury the old hatchet*], for example, it is clear that the adjective *old* combines with the noun *hatchet* syntactically. As an adjective, it has the right formal properties to be in a position between a determiner and a noun. However, from a semantic point of view, it is not compatible with the literal meaning of *hatchet*. It is compatible only with the seemingly idiomatic meaning of *hatchet*, i.e. ‘disagreement/conflict.’ We have to realize here that there is a mismatch between syntactic and semantic behavior in the combination. That is, the combination is “indirect/abnormal” rather than “direct/normal.” In a direct/normal combination, on the other hand, there is no such mismatch between syntactic and semantic behavior. For example, in [*the tall man*], the adjective *tall* modifies the noun *man* not only syntactically but also semantically. The (literal) meaning of *man* is compatible with that of *tall*.

We have to be very careful not to assume that the meaning of ‘disagreement/conflict’ is directly related to the *hatchet* in [*bury the hatchet*]. It comes from the argument of the meaning of the

whole idiom ‘settle/reconcile a conflict/disagreement,’ which can be represented as [SETTLE ([ ], CONFLICT)] formally. This becomes clear with such idioms as [*kick the bucket*] and [*pull the wool over one’s eyes* ‘deceive one’], which can be represented as [DIE ([ ])] and [DECEIVE ([ ], ONE)], respectively. In the former, [*the bucket*] has no (direct) reflections on the whole meaning. In the latter, neither [*the wool*] nor *eyes* have any direct reflections on the meaning. Hence, when we say that *old* in [*bury the old hatchet*] is compatible with “the seemingly idiomatic meaning” of *hatchet*, we mean that it is compatible with the argument of the whole idiomatic meaning, i.e. CONFLICT, rather than with the idiomatic non-literal meaning of *hatchet* itself.

The indirect nature of the combination of idiom-internal adjectives and their host nouns become more evident when the adjectives function as adverbials. In [*bury the official hatchet* ‘settle the conflict officially’], for example, the adjective *official* combines with the noun *hatchet* syntactically.<sup>5</sup> However, from a semantic point of view, it neither combines with the literal meaning of *hatchet* nor the assumed idiomatic meaning of *hatchet* ‘disagreement/conflict.’ As it is not compatible even with the argument of the idiomatic meaning CONFLICT, the combination becomes more different from a regular one. As it does not have any adjectival role semantically, it is “coerced” to perform an adverbial role (with the addition of a semantic adverbializer, which can be regarded as a counterpart of the formal *-ly* ending).<sup>6</sup> Now the adjective can combine with the

<sup>5</sup> Although it has an adverbial function, the word *official* in [*bury the official hatchet*] is still an adjective. It is an adjective because it shows the same syntactic distribution as regular adjectives. Notice that syntactic categories are primarily determined by syntactic distribution. All idiom-internal elements keep their formal identities in our approach, regardless of their functions.

<sup>6</sup> The term “coercion” can be defined as follows (Culicover 2009: 472): “an interpretation that is added to the normal interpretation of a word as a consequence of the syntactic configuration in which it appears.” Typical cases of coercion are exemplified in the sentence [*the ham sandwich over in the corner wants another coffee*], which can be paraphrased as [*the person contextually associated with a ham sandwich over in the corner wants another cup of coffee*] (Culicover & Jackendoff 2005: 227-8; cf. Nunberg 1979,

whole idiomatic meaning or its predicate, i.e. SETTLE.

On the basis of the observations above, we conclude that idiom-internal adjectives and their host nouns do not have direct relationships. Although their combinations are regular formally, the adjective does not combine with its host noun semantically. This means that the adjective is not part of the idiom concerned, and more importantly that the component parts of idioms do not have to have their own separate meanings.

We can conceptualize the licensing of idiom-internal adjectives as follows. Formally, the adjective is licensed as far as it satisfies the morphological and distributional properties required in the position. For example, in [*bury the old/official hatchet*] the words *old* and *official*, as adjectives, satisfy the requirements for being in the position between a determiner and a noun. Under our framework, we just need to leave the NP dominating [*the hatchet*] not enclosed with /.../, to indicate that this idiom allows an internal adjective. Semantically, the adjective is licensed when it has a meaning which is compatible either with an argument of the whole idiomatic meaning or with the whole meaning or its predicate. In the former case, the adjective leads to an adjectival function. In the latter case, on the other hand, it leads to an adverbial function.

As for the semantic licensing of [*bury the old hatchet*], we have to check first whether the meaning of *old*, say OLD, is compatible with an argument of the whole idiomatic meaning, i.e. [SETTLE ([ ], CONFLICT)]. As OLD is compatible with the argument CONFLICT, the whole expression has the meaning of ‘settle an old conflict.’ Now, turning to the semantic licensing of [*bury the official hatchet*], we need to check the compatibility of OFFICIAL with CONFLICT. As this is not a normal combination, there have to be other possibilities. We are assuming that, at this point, the adjective is coerced to have an adverbial meaning. Then, we need to check whether the coerced adverbial meaning of OFFICIAL, say OFFICIALLY,<sup>7</sup> is compatible with the idiomatic

Ward 2004). The underlined parts are coerced interpretations.

<sup>7</sup> We can represent the coerced adverbial meaning of the adjective concerned with a pattern of the following (cf. footnote 10): [in the viewpoint/manner/... of being AdjP]. According to Nicolas (1995: 249), “the most commonly available kind of internal modification is ... viewpoint modification, ... about 85% ...” Then, the expression [*bury the official hatchet*] would be

meaning or its predicate, i.e. SETTLE. As this combination is fine, the whole expression can have the meaning of ‘settle a conflict officially.’ Of course, there would be cases where both types of combinations are possible. In such cases, the expressions concerned would be ambiguous between an adjectival reading and an adverbial reading.

We can see a similar phenomenon of indirect combination in some non-idiomatic phrases:<sup>8</sup>

- (12) a. We [had a quick cup of coffee] before lunch.  
 b. He had to find [a fast road] to get there in time.

The underlined adjectives *quick* and *fast* are positioned between a determiner and a noun and, hence, they are in the right positions. However, they have meanings which cannot be combined with the meanings of their host nouns. The expressions in the square brackets mean roughly ‘drank a cup of coffee quickly’ and ‘a road where we can drive fast,’ respectively. A cup cannot be quick and a road itself, if it is not a moving road, cannot be fast. We can see that the adjectives here are used adverbially (with an appropriate amount of coercion), just like those in idioms. From these examples of indirect combination,<sup>9</sup> we can see that our assumptions about the indirect combination in idioms are not unmotivated.

In this section, we have provided a framework to account for the behavior of idiom-internal adjectives without assuming separate meanings of the parts of idioms. We have seen that the meaning of an idiom containing an internal adjective can be obtained from that of the idiom and that of the modifier compositionally. Although the combination is not direct as in regular phrases, it is not random but follows a general pattern<sup>10</sup> of what

---

interpreted as ‘settle a conflict in the viewpoint/manner of being official.’

<sup>8</sup> The data in (12) were brought up to me by Jeehoon Kim (p.c.).

<sup>9</sup> One might assume that [*have a cup of coffee*] is a kind of idioms, probably due to the “lightness” of the verb *have*. If so, the combination of this idiom and *quick* can be accounted for with the same mechanisms as those for idioms. However, [*a road*] does not have any properties of idioms.

<sup>10</sup> According to Culicover & Jackendoff (2005: 228), there is a consensus in the literature that coerced interpretations are “the product of auxiliary principles of interpretation ... they contribute material that makes the

we call “indirect combination.” Hence, we can conclude that idiom-internal adjectives are not part of the idiom. That is, they should not be a part of the idiom concerned. The only thing we need to do with the idiom is to keep the NP containing the host noun not enclosed with /.../.

## 4 Conclusion

In this paper, we have introduced a framework to represent the syntactic flexibility of idioms. Under this framework, we have examined some phenomena which seem to be accounted for only by assuming separate meanings of the parts of idioms. However, we have shown that we can account for all the phenomena without such an assumption. This is accomplished by positing the set of constructions as a major component of grammar and by capturing the indirect nature of the combination between idiom-internal adjectives and their host idioms.

Focusing on the behavior of idiom-internal adjectives, we have conceptualized a framework to account for the indirectness in the combination of adjectives and their host idioms. By elucidating the nature of this combination, we are absolved from the almost impossible task, especially from a computational point of view, of assigning separate meanings to the component parts of idioms. Consequently, we came to prove that idioms are formally flexible and semantically non-transparent. If we could not figure out that idiom-internal modifiers are not part of idioms, we would not have reached the conclusion that idioms are not transparent/compositional.

## Acknowledgments

I appreciate the comments from the reviewers of PACLIC 28 and PACLIC 29. I also express my sincere appreciations to Jeehoon Kim and the reviewers of BCGL 8 (The 8<sup>th</sup> Brussels Conference on Generative Linguistics: The Grammar of Idioms). This work was supported by the 2015 research fund of Hankuk University of Foreign Studies.

## References

- Chae, Hee-Rahk. 2014. A Representational System of Idiomatic Constructions: For the Building of Computational Resources. *Linguistic Research*, 31: 491-518.

---

sentence semantically well-formed and that plays a role in the sentence’s truth-conditions” (cf. Jackendoff 1997).

- Choi, Kyeong Bong. 2014. *A Dictionary of Korean Idioms* [written in Korean]. Ilchokak.
- Croft, William, and D. Alan Cruse. 2004. *Cognitive Linguistics*. Cambridge Univ. Press.
- Culicover, Peter. 2009. *Natural Language Syntax*. Oxford Univ. Press.
- Culicover, Peter, and Ray Jackendoff. 2005. *Simpler Syntax*. Oxford Univ. Press.
- Espinal, M. Teresa, and Jaume Mateu. 2010. On Classes of Idioms and Their Interpretation. *Journal of Pragmatics*, 42: 1397-1411.
- Everaert, Martin, Erik-Jan van der Linden, Andre Schenk, and Rob Schreuder, eds. 1995. *Idioms: Structural and Psychological Perspectives*. Lawrence Erlbaum Associates, Inc.
- Fillmore, Charles J., Paul Kay and Mary C. O'Connor. 1988. Regularity and Idiomaticity in Grammatical Constructions: The Case of *let alone*. *Language*, 64: 501-538.
- Geeraerts, Dirk. 1995. Specialization and Reinterpretation in Idioms. In Everaert et al. (1995).
- Gibbs, Raymond W., Jr. 1995. Idiomaticity and Human Cognition. In Everaert et al. (1995).
- Gibbs, Raymond W., Jr. 2007. Idioms and Formulaic Language. In Dirk Geeraerts and Hubert Cuyckens, eds. *Oxford Handbook of Cognitive Linguistics*. Oxford Univ. Press.
- Goldberg, Adele E. 1995. *Constructions: A Construction Grammar Approach to Argument Structure*. Univ. of Chicago Press.
- Jackendoff, Ray. 1997. *The Architecture of the Language Faculty*. MIT Press.
- Nicolas, Tim. 1995. Semantics of Idiom Modification. In Everaert et al. (1995).
- No, Yongkyoon. 2002. *A Dictionary of Basic Idioms on Grammatical Principles* [written in Korean]. Hankookmunhwasa.
- Nunberg, Geoffrey. 1978. *The Pragmatics of Reference*. Indiana Univ. Linguistics, Bloomington.
- Nunberg, Geoffrey. 1979. The Nonuniqueness of Semantic Solutions: Polysemy. *Linguistics and Philosophy* 3: 143-184.
- Nunberg, Geoffrey, Ivan Sag, and Thomas Wasow. 1994. Idioms. *Language*, 70: 491-538.
- Sag, Ivan, Timothy Baldwin, Francis Bond, Ann Copestake, and Dan Flickinger. 2002. Multiword Expressions: A Pain in the Neck for NLP. In Alexander Gelbuk, ed. *Computational Linguistics and Intelligent Text Processing: Third International Conference (CICLing-2002)*. Springer-Verlag.
- Schenk, Andre. 1995. The Syntactic Behavior of Idioms. In Everaert et al. (1995).
- Ward, Gregory. 2004. Equatives and Deferred Reference. *Language* 80: 262-289.
- Wasow, Thomas, Ivan Sag, and Geoffrey Nunberg. 1983. Idioms: An Interim Report. In S. Hattori and K. Inoue, eds. *Proceedings of the XIIIth International Congress of Linguists*. CIPL, Tokyo.
- Wulff, Stefanie. 2013. Words and Idioms. In Thomas Hoffmann and Graeme Trousdale, eds. *The Handbook of Construction Grammar*. Oxford Univ. Press.

# Asymmetries in Scrambling and Distinctness of Copies

Gwangrak Son

English Language and Literature  
Kyungpook National University  
Daegu, South Korea 702-701  
gson@knu.ac.kr

## Abstract

This paper is an investigation of LF-copies created by scrambling in the context of FNQ-constructions. It demonstrates that movement leaves a copy at LF only when it targets a position within the next search space; it does not leave an LF copy if movement takes place too close within a single domain of search space. By characterizing this in terms of “Distinctness of Copies,” this paper provides a principled account to all structural variations that have posed substantial problems in previous approaches.

## 1 Introduction

It has been widely noted that the extraction of different arguments can be subject to different restrictions. One such case involves the distribution of floating numeral quantifiers (FNQs) in Japanese and Korean.<sup>1</sup> In these languages, extraction of objects licenses an associated FNQ, while that of subjects does not (Kuroda, 1980; Saito, 1985; Ko, 2007; Miyagawa and Arikawa, 2007; Miyagawa 2013; J. Kim, 2013, among many others). Although this understanding about the subject-object asymmetry is well-established in the literature, it is not always easy to reach a conclusion about the grammaticality of the sentences that contain subject FNQs. We also find that the Locality approach (Saito, 1985; Miyagawa, 1989; 2010; Miyagawa and Arikawa, 2007, etc.), the most compelling account for this exciting but bewildering phenomenon, is not entirely acceptable because of its shortcomings in terms of either empirical coverage or the explanatory power essential for theories in modern lin-

<sup>1</sup> Sportiche’s (1988) proposal for a theory of floating quantifiers relies on two independently motivated assumptions: (i) a quantifier and its associate NP are generated under a single constituent, and (ii) the NP moves up for a number of reasons while stranding the quantifier in its base-generated position. I hold these assumptions throughout this paper.

guistic research.

In this paper, we lay down two minimalist assumptions and demonstrate that simply by combining these, all the lingering problems germane to the previous approaches (including the Locality) can be eliminated. Additionally, a variety of puzzles that arise in scrambling contexts all fall out nicely. The two hypotheses include Chomsky’s (2000; 2001; 2008) PIC (Phase Impenetrability Condition) and a novel proposal of DC (Distinctness of Copies), which is an elaboration on Richards’s (2000; 2010) principle of Distinctness. Insofar as the current analysis is sustained, it will then supply empirical evidence in support of these theoretical assumptions in the minimalist program, while further clarifying some residual problems.

The structure of the paper is as follows. In section 2, while reviewing the Locality approach to FNQs, we tease out an important fact that structural variations of FNQ-constructions are contingent on the availability of LF copies created by scrambling. In section 3 we lay down our proposals, and in section 4 we demonstrate that the DC, in conjunction with the PIC, provide a principled and unitary account to all the structural variations that have posed substantial problems in previous approaches. Section 5 is a conclusion of the paper, with a discussion of some predictions that follow from the current analysis.

## 2 Locality and Problems

Since it was first observed by Haig (1980) and Kuroda (1980), the subject-object asymmetry of FNQs, shown in (1) in Japanese, has been described by the term “Locality,” defined in terms of mutual c-command between an NP (or an NP trace) and its associated numeral quantifier.<sup>2</sup>

<sup>2</sup> Locality:

- a. The NQ and its associated NP observe strict locality (Saito, 1985).
- b. The NQ or its trace and the NP or its trace must mutually c-command each other (Miyagawa, 1989).

- (1) a. \*Gakusei-ga sake-o san-nin nonda.  
 student-Nom sake-Acc 3-CL<sub>subj</sub> drank  
 ‘Three students drank sake.’  
 b. Sake-o gakusei-ga san-bon nonda.  
 Sake-Acc student-Nom 3-CL<sub>obj</sub> drank  
 ‘Students drank three bottles of sake.’

In the era of Government and Binding, it was assumed that a subject cannot scramble, as indicated by Saito’s (1985) “ban on subject scrambling,” and is merged directly in its surface position. Since the subject does not involve movement, it has no  $\nu$ P-internal trace, resulting in violation of the Locality requirement in sentence (1a). In contrast, an object is assumed to scramble freely and leaves a trace. Consequently, the trace and its associated NQ in VP satisfy the required constraint, leading to the grammaticality of sentence (1b). In this view, the subject-object asymmetry of FNQs in scrambling contexts comes as a consequence of the trace visibility in a position next to the NQs. (2) below depicts this account under the Locality approach.

- (2) a. \*Gakusei-ga sake-o [NO TRACE san-nin] nonda.  
 b. Sake-o gakusei-ga [TRACE san-bon] nonda.

As a reader might already have observed, this account can hardly hold in its original form in the minimalist program, one major finding of which is that the subject is derived from its  $\nu$ P-internal position (Kitagawa, 1986; Sportiche, 1988; Kuroda, 1988; Koopman and Sportiche, 1991, etc.). Under the so-called VP-Internal Subject Hypothesis (VPISH), (2a) could have the following structure, in which the subject has scrambled over the preposed object from its lower base position (Bobaljik, 2003:115, see also Bošković, 2004).

- (3) Gakusei-ga sake-o [<sub>VP</sub> t<sub>subj</sub> san-nin [<sub>VP</sub> t<sub>obj</sub> nonda]]

This structure, once its validity is proven, will significantly weaken the Locality approach since it obliterates the disparate patterns of the traces between subject and object. However, one might argue, in full compliance with Saito’s original intuition of the “ban on subject scrambling,” that the “double-scrambling structure” (3) is less economic than (2a) since it contains more movement steps to arrive at the same word order. Therefore, from the economy perspective, the Locality account still holds that (2a) is an optimal structure and that there is no licensing trace for the stranded subject NQ. If we strictly adhere to this view, the prediction is clear: there should be no stranded quantifier associated with the subject.

Unfortunately, however, this prediction is too general, since in the literature we find a number of counterexamples where subject NQs occur precisely in this structural format, yet maintain grammatical integrity.<sup>3</sup> See Kuno, 1973; Ishii, 1998; Takami, 1998; Gunji and Hasida, 1998; Kuno and Takami, 2003; Nishigauchi and Ishii, 2003; Yoshimoto et al., 2006; Miyagawa and Arikawa, 2007; Miyagawa 2010; 2013 for Japanese examples illustrating this fact; see also Lee, 2003; S. Kim, 2004; Moon, 2007; Y. Kim, 2008; J. Kim, 2013, and Son, 2015 for the same fact in Korean.

A further complication arises with the Locality analysis. Miyagawa (2001; 2003; 2005) has argued that Japanese does exhibit EPP effects, and a scope contrast as described below comes as a consequence of EPP-movement by either the subject or the object to a position higher than negation.

- (4) a. Zen’in-ga sono tesuto-o uke-nakat-ta.  
 all-Nom that test-Acc take-Neg-Past  
 ‘All did not take that test.’  
 \*not > all, all > not  
 b. Sono tesuto-o zen’in-ga t uke-nakat-ta  
 that test-Acc all-Nom take-Neg-Past  
 ‘That test, all did not take.’  
 not > all, (all > not)

On Miyagawa’s account, T bears a strong EPP-feature in Japanese, and hence it requires movement of some NP to [Spec,TP] in overt syntax; in (4a), the Spec of TP is oc-

<sup>3</sup> The examples of (i) and (ii) below are representative of the nonstandard paradigms (i.e. exceptions to standard paradigms) in Japanese and Korean, respectively.

- (i) a. ?Gakusei-ga sake-o [PAUSE] san-nin nonda.  
 students-Nom sake-Acc 3-CL<sub>subj</sub> drank  
 ‘Three students drank beer.’ (M&A:651)  
 b. Gakusei-ga watasi-no hon-o futa-ri-sika kaw-ana-  
 student-Nom my-Gen book-Acc 2-CL<sub>subj</sub>-only buy-Neg-  
 katta.  
 Past  
 ‘Only two students bought my book.’  
 (Takami, 1998:92)
- (ii) a. Marathon juja-deul-i kyeolseungjeum-ul  
 Marathon runner-PL-Nom finishing line-Loc  
 taseos-myeong thongkwahaessta.  
 5-CL<sub>subj</sub> pass-Pst  
 ‘Five marathon runners have passed the finishing line.’  
 b. Haksaeng-tul-i sukje-lul jikeumkkaji  
 student-PL-Nom homework-Acc so far  
 se-myeong jechulhaesseo  
 3-CL<sub>subj</sub> submitted  
 ‘Three students submitted homework so far.’  
 (Son, 2015: 232-239)

occupied by the subject, while in (4b) it may be occupied by either the subject or the scrambled object. Crucially, in (4b), the subject can remain *in-situ* in the specifier position of  $vP$ , where it may be interpreted within the scope of negation. If the subject could be externally merged in [Spec, $vP$ ] as in (4b), and if we imagine that the higher subject in (4a) is indeed the one derived from the lower  $vP$ -internal position through scrambling (in compliance with the VPISH),<sup>4</sup> the double-scrambling structure of (3) cannot simply be banished by economy, because the movement operation of the subject is a bona fide fact. This consideration, then, brings us back to the initial quest regarding the subject-object asymmetry in (1) since in this view both NPs are permitted to scramble to TP and leave traces alike.

This state of affairs seems to indicate that the Locality account is now obsolete. Alternatively, it could mean that both structures (2a) and (3) coexist in Japanese and Korean grammar, in a way suggested by Miyagawa and Arikawa (2007) (see also Miyagawa, 2010), so each may represent the standard versus nonstandard case (i.e. exceptions to the standard paradigm) of the FNQ-constructions. Although Miyagawa and Arikawa (2007) (M&A, hereafter) have merely suggested this (based on phonological experiments), we will show that they are indeed correct. We will support them by providing a syntactic ground concerning the varying structures of (2a) and (3) for subject scrambling and (2b) for object scrambling. More specifically, we claim that a subject undergoes scrambling by either (2a) or (3), yielding a disparate LF structure. The subject lacks an LF trace (or copy, in minimalist terms) in the former, but leaves it in the latter derivation. The object, on the other hand, always leaves an LF trace after scrambling, as illustrated by (2b). We claim that these varia-

<sup>4</sup> Based on such examples as the following, Ko (2007:5) claims that subject scrambling is indeed possible in Korean.

- (i) a. John-*i* [<sub>CP</sub> na-nun [<sub>CP</sub> *t*<sub>i</sub> Mary-lul mannassta-ko] sayngkakhanta]  
 J-Nom I-Top M-ACC met-C think  
 ‘John, I think that *t* met Mary.’  
 b. Haksayng-tul-*i* pwunmyenghi *t*<sub>i</sub> sey-myeng maykcwu-lul  
 student-PL-Nom evidently 3-CL<sub>subj</sub> beer-Acc  
 masiessta  
 drank  
 ‘Evidently, three students drank beer.’

In (ia) above, an embedded subject has scrambled over a matrix subject, and in (ib) it is even separated from its NQ by a sentential adverb. See Kurata, 1991; Lee, 1993 and Sohn, 1995 for more examples of this sort in Korean and Japanese.

tions of LF copies follow from two minimalist assumptions that we lay down in section 3.

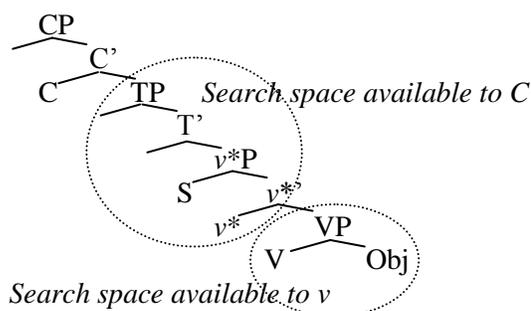
### 3 Proposals

An important development of the past decade is the hypothesis that syntactic operations are not optional but triggered (i.e. the Last Resort principle). The most influential work along this line is Chomsky’s (1993, et seq.) proposal that uninterpretable features play a central role in the triggering process under the Phase Impenetrability Condition (PIC). In this system, only phase heads (C and transitive  $v^*$ ) bear uninterpretable features, and consequently, phase-internal elements are forced to move only through phase edges. The phase heads also mark points in the derivation at which the complements of the phase heads are transmitted from narrow syntax to the interfaces, PF and LF. Once the selected structure undergoes Transfer to the interface components, its phonological and semantic information is no longer accessible for further operations. Chomsky’s (2000) formulation of the PIC is given in (5); the resultant patterns of search spaces are depicted in (6).

- (5) Phase Impenetrability Condition (Chomsky, 2000:108)

In phase  $\alpha$  with head H, the domain of H is not accessible to operations outside  $\alpha$ ; only H and its edges are accessible to such operations.

- (6) Search spaces of phase heads



As shown in (6), the search space of the phase head  $v^*$  is VP that contains V and the object. On the other hand, the higher phase C has Spec-T, T, Spec- $v^*$ , and  $v^*$  in its search space, to the exclusion of VP. Since the PIC in (5) imposes VP-Transfer as soon as  $v^*P$  is complete, the probe C cannot look inside VP. In other words, the VP and

any elements contained therein are no longer accessible to the phase head C (and the head T, which becomes a probe due to C). The search spaces sketched in (6) will have a direct impact on the distribution of FNQs in Japanese and Korean, as will become clear shortly.

Along with this, based on Richards's (2000; 2010) principle of Distinctness, we elaborate another constraint that holds presumably in narrow syntax, i.e. before a derivation reaches PF- and LF-interfaces. On PF side, there is a general tendency to reduce or eliminate phonological "redundancy" within a certain minimal domain, similar to the effects of the OCP in phonology. Analogous phenomena are also found in narrow syntax, among which Richards's principle of Distinctness on linearization is particularly instructive to us in its scope and effects.<sup>5</sup> The Distinctness principle states that two nodes that are too similar, e.g., of the same category, cannot be in the same phase domain.

(7) Distinctness Principle (Richards, 2010)

If a linearization statement  $\langle \alpha, \alpha \rangle$  is generated, the derivation crashes.

Although Richards's (2000) principle of this only makes use of node labels and does not refer to particular information of lexical items on terminals, we may further assume that linear ordering is indeed sensitive to the phonetic forms on terminals, not just to their categorical nodes. One compelling piece of evidence for this direction comes from Grohmann's (2003) Condition on Domain Exclusivity (CDE) in (8), which uses phonetic information of the syntactic objects on terminals, while taking precisely the same effect as Distinctness.

(8) Condition on Domain Exclusivity (Grohmann, 2003:272)

An object O in a phrase marker must have an exclusive occurrence in each Prolific Domain  $\Pi\Delta$ , unless duplicity yields a drastic effect on the output; that is, a different realization of O in that  $\Pi\Delta$  at PF.

The CDE in (8) permits only one instance of the same phonetic expression in a particular syntactic domain, namely, Prolific Domain (PD) in his terms, to the effect that there would be no two copies of phonetically identical form within a PD. This explains why such an example as (9a) below, in distinction from (9c), is ungrammatical. For convergence, one instance of the copies (i.e. the lower one) must be spelled out in a distinct phonetic form, as in (9c).

(9) a. \*John likes John.

b. [<sub>VP</sub> John v [<sub>VP</sub> likes John]]

c. John likes himself.

(Grohmann, 2003:275)

Importantly, note at this point that Grohmann's CDE is reducible to Distinctness once we make the latter applicable to the set of phonetically identical copies in the course of syntactic computations. In this view, multiple occurrences of the same phonetic form cannot be linearized in syntax because doing so creates an indistinguishable set within a relevant domain. Following this line of reasoning and taking Chomsky's search spaces in (6) to be the relevant domain where Distinctness applies, we propose the following generalization.

(10) Distinctness of Copies (DC)

Identical copies cannot appear within a search space (defined under the PIC).

The essence of the Distinctness of Copies (DC) is to ban phonetically identical copies from occurring within a single search space. In Grohmann's system, this condition is met by the operation of Copy Spell-Out (within a PD), i.e., by spelling out a lower copy in a distinct phonetic form. However, crucially, there is another way of satisfying this condition: By simply "deleting" one party of the two-membered chain. We contend that this is exactly what happens in the course of syntactic derivations involving subject- and object scrambling in the context of FNQ-constructions. When movement takes place within a search space, a copy in the tail—as is usual in the process of copy-deletion (Nunes, 2001)—is wiped out in deference to the DC. We call this operation in (11) Copy Elimination. Similar to Grohmann's Copy Spell-Out, this operation is a Last Resort strategy to fulfill the re-

<sup>5</sup> For more work on "syntactic OCP," see Mohan (1994), Yip (1998), Anttila and Fong (2001), Erlewine (2013), and the references cited therein.

quirement imposed by the DC, by turning a two-membered chain into a single-membered chain.

#### (11) Copy Elimination

If a movement chain  $\langle \alpha, \alpha \rangle$  is created within a search space, eliminate the lower copy.

Since we assume that the Copy Elimination in (11) holds in narrow syntax before a derivation reaches the PF- and LF-interfaces, as is in Richards’s Distinctness and Grohmann’s CDE, the consequence of the operation is formidable, especially in its effects on LF.<sup>6</sup> The subsequent section is a demonstration of how the proposed principle of the DC, in conjunction with Chomsky’s PIC, correctly predicts the bewildering patterns of the copies noted in section 2; the subject-object asymmetry in (2a) versus (2b), and the standard-nonstandard variations of subject scrambling in (2a) versus (3).

## 4 Analyses

By adhering to the essence of the Locality approach, we assume in this article that an NQ must be in a strict local relation with its host NP for interpretation. However, deviating from major works in this approach (Saito, 1989; Miyagawa, 1989; 2001; 2013 and M&A), we adopt the minimalist assumption of the VPISH (Kitagawa, 1986; Sportiche, 1988; Kuroda, 1988; Koopman and Sportiche, 1991, etc.). That is, a subject is externally merged in the Spec of  $\nu P$  regardless of the standard and nonstandard variations of subject scrambling. This implies that M&A’s (2a) and (3), which represent the structure of the standard and nonstandard paradigms, respectively, are indistinctive as the subject is commonly originated from the  $\nu P$ -internal position. They share an identical structure in (12).

(12) Gakusei-ga sake-o [ $\nu P$  t<sub>subj</sub> san-nin [ $\nu P$  t<sub>obj</sub> nonda]]

Given the common structure of (12) for both paradigms, the judgmental variations between (2a) and (3) now turn out to be contingent on the avail-

<sup>6</sup> Since Richards’s (2010) Distinctness is sensitive to the distribution of strong phase boundaries, it is obviously not a pure PF-operation. In the same vein, since Grohmann’s (2003) CDE makes use of Prolific Domains within the sphere of narrow syntax, it also cannot be viewed purely as a PF-operation.

ability of the subject traces *in-situ*. That is, if the subject trace in the [ $\text{Spec}, \nu P$ ] is somehow made “invisible” and hence the structure looks like the standard paradigm of (2a), the stranded subject NQ will be left uninterpretable since no licensing DP is available next to it. On the other hand, if the *in-situ* subject is “visible” and available for interpretation of the adjoining  $\text{NQ}_{\text{subj}}$ , the sentence improves its grammaticality.<sup>7</sup> This constitutes a nonstandard case of subject scrambling, as depicted in (3). On this reasoning, an emerging question is how to explain the availability of the traces that have a direct impact on the interpretability of the FNQs. Chomsky’s PIC and our novel proposal of the DC provide an adequate answer to this question.

First, consider (13), a structure of the standard paradigm built on this view. [From now on, we use “copy” in place of “trace” in favor of minimalist terms.]

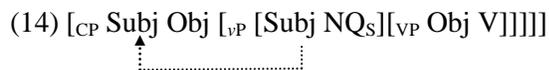
(13) [ $\text{TP}$  Subj Obj [ $\nu P$  [Subj  $\text{NQ}_S$ ][ $\nu P$  Obj V]]]]

In the above, the subject raises from its  $\Theta$ -position in [ $\text{Spec}, \nu P$ ] to [ $\text{Spec}, \text{TP}$ ], driven by the EPP-feature on T (Miyagawa, 2001; 2003; 2005). Crucially, the two copies of the movement chain,  $\langle \text{Spec-T}, \text{Spec-}\nu \rangle$ , are both contained in the search space of C that covers Spec-T, T, Spec- $\nu$ , and  $\nu$  (see (6)). Since this chain does not comply to the principle of the DC in (10), the lower copy in [ $\text{Spec}, \nu P$ ] undergoes Copy Elimination. The stranded subject NQ then fails to meet the Locality requirement at LF, causing a problem with its interpretation.

Although the standard derivation (13) crashes for the aforementioned reason, there is an alternative way of deriving the surface word order of (12). If we take Chomskian style A’-movement that raises an *in-situ* subject to [ $\text{Spec}, \text{CP}$ ] in one fell swoop, as depicted in (14) below (Chomsky, 2001; 2008; see also Pesetsky and Torrego, 2001 and Erlewine, 2013), an interesting result emerges.<sup>8</sup>

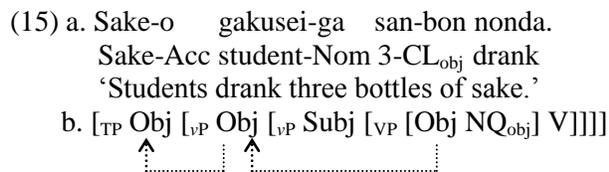
<sup>7</sup> Nonstandard examples are less than perfect in general. They become fully acceptable only with the help of a peculiar sort of prosody around the sentence. See M&A and Son (2015), which independently contend that these peculiar prosodies are what accounts for the degradedness of the nonstandard cases.

<sup>8</sup> Pesetsky and Torrego (2001) have claimed that the subject may check the EPP on C via a direct movement to [ $\text{Spec}, \text{CP}$ ]. On the other hand, Erlewine (2013), based on the Agent Focus



In the A'-movement configuration above, the displacement chain of the subject, <Spec-C, Spec-v>, obeys the DC as the copy in the head stays outside the search space of C. As a result, the *in-situ* subject copy passes down to the interfaces and supports its adjoining NQ at LF.<sup>9</sup> This explains how an otherwise ungrammatical sentence is rendered “saved” through the nonstandard derivations of subject scrambling.

Let us now proceed to see how the current proposal successfully captures the conventional subject-object asymmetry in scrambling. Recall that an object NQ can be freely separated from its associated NP by a subject or any other elements in a sentence. This is in contrast with the pattern of a subject NQ that only allows such separation optionally, resulting in varying judgments as we have seen. From the perspective developed in this article, the source for this asymmetry is surprisingly simple. Consider the following example of object scrambling, repeated from (1b), with its derivation in (15b).



As depicted in (15b), the object raises to [Spec,TP] in a successive cyclic fashion; it first moves to the outer edge of vP and further scrambles to the Spec of TP for the purpose of the EPP. Of these, the first step of displacement, <Spec-v, Complement of V>, satisfies the DC as the copy in the position of tail is the only expression of the object in the search space of v\*, namely, VP (see (6)). Consequently, the *in-situ* copy transfers and becomes visible at LF, licensing its adjoining NQ at the interpreta-

phenomenon in the language of Kachickel, argues that the EPP is not required in this language. As such, the subject is allowed to move to [Spec,CP] without stopping over in the specifier position of TP.

<sup>9</sup> A warning is in order here. Although the subject chain in (14) is consistent with the DC and leaves an interpretable copy at LF, it does not necessarily mean that the copy is visible at PF. This is because the PF-interface has an independent process of copy-deletion, in a way suggested by Nunes (2001) and Corver and Nunes (2007).

tional level. Note that the second step of movement, which has the head in the [Spec,TP] and the tail in the [Spec,vP], contravenes with the DC in the search space of C. However, the concomitant deletion operation in the tail exerts no impact on the interpretability of the object NQ since it has already undergone Transfer and becomes interpretable by the help of the string-adjacent object copy *in-situ*. The possible separation of the object NQ from its host NP is thus accounted for.

In fact, since the object merges with V and undergoes Transfer independently of its higher copy upon VP-Transfer, it is invariably predicted to be visible at LF. As Abels (2003) has correctly stated by his Anti-Locality, VP-internal movement, e.g. from the complement position to the specifier position of VP, is prevented. As such, whether it moves to [Spec,TP] or [Spec,CP] via A- or A'-movement, it always leaves an interpretable copy at LF. This is in contrast with the subject, the chain link of which may or may not leave an LF copy; it leaves a copy if it targets an A-position in [Spec,TP], but not if it moves directly to [Spec,CP] via topicalization. This provides a source of the asymmetry between the subject and object with regard to the interpretability of the FNQs associated with them.

## 5 Conclusion

In this paper, we have seen that the operation of the DC is quintessential in determining the availability of the copies at LF in a lower position of a two-membered chain. Subject scrambling from [Spec,vP] to [Spec,TP] lacks an interpretable copy in the tail position, while the same movement to [Spec,CP] does leave such a copy. On the other hand, the object always leaves a copy at LF after scrambling. These variations turn out to be a result of interactive operations of the DC with the PIC. Since the DC demands an exclusive copy of the same expression in a search space of the PIC, movement leaves a copy at LF only when it targets a position within the next search space; it does not leave an LF copy if movement takes place too close within a single domain of search space. We may refer to this dependency as the “Semantic Copy Effect.”

- (16) The Semantic Copy Effect  
 Movement leaves a copy at LF for semantic interpretations only when it targets a position

within the next search space (although the copy may be deleted on the PF side).

Overall, the current analysis makes the following predictions:

(17) Predictions:

- A. A- chain of the subject, <Spec-T, Spec-v>, lacks a copy *in-situ* at LF.<sup>10</sup>
- B. A'- chain of the subject, <Spec-C, Spec-v>, leaves a copy *in-situ* at LF
- C. The object always leaves a copy at LF, whether it undergoes A- or A'-movement.
- D. An unaccusative/passive subject will pattern like the object and leave a copy *in-situ*, while an unergative subject may or may not leave a copy at LF.<sup>11</sup>
- E. An A'-moved subject (i.e. nonstandard paradigms) will have a topic interpretation.<sup>12</sup>

<sup>10</sup> This prediction has a direct bearing on Chomsky's (1995) claim of "No A-movement traces (or copies)." This article shows that Chomsky does not provide the whole picture. It is not that the copies never existed, but that previously manifesting copies were deleted by the operation of Copy Elimination. The proposal of the DC explains why in the case of objects with A-movement, copies still remain at LF, as stated in (17C). Further investigation is needed to see if this remains consistent in other languages.

<sup>11</sup> The following examples demonstrate that this prediction holds true in Korean (data adapted from Ko, 2007:68). See Miyagawa, 1989; Mihara, 1998; Kuno and Takami, 2003; M&A; S. Kim, 2004; and J. Kim, 2013 for more examples of this kind in Korean and Japanese.

- (i) a. Koyangi-ka pyeong-ulo sey-mali juk-ess-ta (unaccusative)  
cat-Nom disease-by 3-CL<sub>animal</sub> die-Past-Dec  
'Three cats died from diseases.'
- b. Eoje, catongcha-ka koyhan-eykey two-tay pusu-eoji-ess-  
yesterday, car-Nom robber-Dat 2-CL<sub>car</sub> break-Pass-  
ta (passive)  
Past-Dec  
'Yesterday, two cars were broken into by a robber.'
- c. ?\*Haksayng-tul-i kaki-tul ton-ulo two-myeong cenhwaha-  
student-PL-Nom self-PL money-by 2-CL telephone-  
yess-ta (unergative)  
Past-Dec  
'Two students telephoned with their own money.'

On the other hand, for the external merge position of the unaccusative/passive subject (i.e. a complement position of V), in distinction from that of the unergative subject, see Perlmutter, 1978; Belletti and Rizzi, 1981; Burzio, 1986; Miyagawa, 1989; Hale and Keyser, 1993, and Chomsky, 1995.

<sup>12</sup> Lee (2003; 2006), S. Kim (2004), J. Kim (2013) and Son (2015) have independently claimed that the so-called non-standard examples are motivated by the information structure, and carry a discourse/pragmatic meaning of a topic-comment

Some of these predictions have been proved empirically in natural languages; some others remain yet unexplored. Although we have drawn these predictions through the study of scrambling phenomenon in the context of FNQ-constructions in Japanese and Korean, we wish to see their validity in other domains of movement and in other languages as well. With much anticipation for research towards this direction, we conclude this paper.

## References

- Abels, Klaus. 2003. *Successive Cyclicity, Anti-locality, and Adposition Stranding*. Doctoral dissertation. University of Connecticut.
- Anttila, Arto, and Vivienne Fong. 2001. The Partitive Constraint in Optimality Theory. *Journal of Semantics*, 17:281-314.
- Bobaljik, Jonathan David. 2003. Floating Quantifiers: Handle with Care. In *The second GLOT International state-of-the-article book*, ed. Lisa Cheng and Rint Sybesma, 107-148. Berlin: Mouton de Gruyter.
- Bošković, Željko. 2004. Be Careful Where You Float Your Quantifiers. *Natural Language and Linguistic Theory*, 22: 681-741.
- Chomsky, Noam. 1993. A Minimalist Program for Linguistic Theory. In *The View from Building 20: Essay in Linguistics in Honor of Sylvain Bromberger*, ed. Ken Hale and Samuel J. Keyser. Cambridge: MIT Press.
- Chomsky, Noam. 1995. Categories and Transformations. In *The Minimalist Program*, 219-394. Cambridge, MA: MIT Press.
- Chomsky, Noam. 2000. Minimalist Inquiries: The Framework. In *Step by step: Essays on Minimalist Syntax in Honor of Howard Lasnik*, ed. R. Martin, D. Michaels, and J. Uriagereka, 89-155. Cambridge, MA, MIT Press.
- Chomsky, Noam. 2001. Derivation by Phase. In *Ken Hale: A Life in Language*, 1-52. ed. Michael Kenstowicz. Cambridge: MIT Press.
- Chomsky, Noam. 2008. On Phases. In *Foundational issues in Linguistic Theory*, 133-166. ed. Freiden Robert, Carlos P. Otero and Maria Luisa Zubizarreta. Cambridge, Mass.: MIT Press.
- Corver, Nibert and Jairo Nunes. 2007. *The Copy Theory of Movement*. John Benjamins Publishing Company.
- Erlewine, Michael Yoshitaka. 2013. Anti-locality and Kaqchikel Agent Focus. To appear in *Proceedings of*

or background-focus. I would like to refer the reader to these works for examples and detailed arguments.

- the 31st West Coast Conference on Formal Linguistics*, 31. Cascadilla Press.
- Grohmann, Kleanthes K. 2003. Successive Cyclicity under (Anti-)local Considerations. *Syntax*, 6(3):260-312.
- Gunji, Takao and Koiti Hasida. 1998. Measurement and quantification. In *Topics in constraint-based grammar of Japanese*, ed. Takao Gunji and Koiti Hasida, 39-79. Dordrecht: Kluwer.
- Haig, John. 1980. Some Observations on Quantifier Floating in Japanese. *Linguistics*, 18:1065-1083.
- Ishii, Yasuo. 1998. Floating Quantifiers in Japanese: NP quantifiers, VP quantifiers, or both? *Researching and Verifying on Advanced Theory of Human Language*, 149-171. Graduate School of Language Sciences Kanda University of International Studies, Japan.
- Kim, Jong-Bok. 2013. Floated Numeral Classifiers in Korean: A Non-Derivational, Functional Account Floating Quantifiers. *Lingua*, 133:189-212.
- Kim, Soo-Yeon. 2004. Constraints on Distributional Patterns of Floating Quantifiers. In *Eoneuhak [The Linguistic Association of Korean Journal]*, 38:43-66.
- Kim, Yong-Ha. 2008. Against Cyclic Linearization: Scrambling and Numeral Quantifiers in Korean. In *Language Research*, 44(2):241-274.
- Kitagawa, Yoshihisa. 1986. *Subjects in English and Japanese*. Doctoral dissertation. University of Massachusetts.
- Ko, Heejeong. 2007. Asymmetries in Scrambling and Cyclic Linearization. *Linguistic Inquiry*, 38(1): 49-83.
- Koopman, Hilda and Dominique Sportiche. 1991. The Position of Subjects. *Lingua*, 85(2): 211-58.
- Kuno, Susumu. 1973. *The Structure of the Japanese Language*. Cambridge, Mass.: MIT Press.
- Kuno, Susumu, and K. Takami. 2003. Remarks on Uu-accusativity and Unergativity in Japanese and Korean. In *Japanese/Korean Linguistics*, 12:280-294. ed. William McClure. Stanford. Calif.: CSLI.
- Kurata, Kiyoshi. 1991. *The Syntax of Dependent Elements*. Doctoral dissertation. University of Massachusetts, Amherst.
- Kuroda, Sige Yuki. 1980. Bunkoozo-no Hikahu. In *Nitieigo Hikakoozo: Bunpoo*, ed. T. Kunihiko. Tokyo, Takkushan.
- Kuroda, Sige Yuki. 1988. Whether We Agree or Not: A comparative syntax of English and Japanese. *Linguistic Investigations*, 12:1-47.
- Lee, Chungmin, 2003, Contrastive Topic and/or Contrastive Focus. In *Japanese/Korean Linguistics*, 12:382-420. ed. William McClure. Elsevier.
- Lee, Chungmin, 2006. Contrastive Topic/Focus and Polarity in Discourse. In *Where Semantics Meets Pragmatics*, ed. K. von Stechow, 382-420. Elsevier.
- Lee, Young-Suk. 1993. *Scrambling as Case-driven Obligatory Movement*. Doctoral Dissertation. University of Pennsylvania, Philadelphia, Penn.
- Mihara, Ken-ichi. 1998. *Nihongo-no Toogo Koozo [Syntactic structures in Japanese]*. Tokyo: Syoakusya.
- Miyagawa, Shigeru. 1989. *Structure and case-marking in Japanese*. New York: Academic Press.
- Miyagawa, Shigeru. 2001. The EPP, Scrambling, and Wh-in Situ. Chapter 9. In *Michael Kenstowicz*. ed. by Ken Hale. The MIT Press.
- Miyagawa, Shigeru. 2003. A-movement Scrambling and Options Without Optionality. *Word order and Scrambling*, ed. S. Karimi, 177-200. Oxford: Blackwell.
- Miyagawa, Shigeru. 2005. EPP and semantically vacuous scrambling. *The Free Word order Phenomenon*, ed. J. Sabel and M. Saito, 181-220. Berlin: Mouton de Gruyter.
- Miyagawa, Shigeru. 2010. *Why Agree? Why Move? Linguistic Monograph Fifty-Four*. The MIT Press, Cambridge, Massachusetts.
- Miyagawa, Shigeru. 2013. Telicity, Stranded Numeral Quantifiers, and Quantifier Scope. Chapter 2. Ms. MIT.
- Miyagawa, Shigeru and Koji Arikawa. 2007. Locality in Syntax and Floating Numeral Quantifiers. *Linguistic Inquiry*, 38(4): 645-670.
- Mohanan, Tara. 1994. Case OCP: A Constraint on Word Order in Hindi. In *Theoretical Perspectives on Word Order in South Asian languages*, ed. Miriam Butt, Tracy Holloway King, and Gillian Ramchand, 185-216. Stanford, CA: CSLI.
- Moon, Gui-Sun. 2007. Scrambling and PIC: Against Cyclic Linearization. *Studies in Generative Grammar*, 17(2):233-252.
- Nishigauchi, Taisuke and Yasuo Ishii. 2003. *Eigo Kara Nihongo o Miru [Looking at Japanese from English]*. Tokyo: Kenkyusha.
- Nunes, Jairo. 2001. Sideward Movement. *Linguistic Inquiry*, 32:303-344.
- Pesetsky, David and Esther Torrego. 2001. T-to-C Movement: Causes and Consequences. In *Ken Hale: A life in language*, ed. Michael Kenstowicz. Cambridge, Mass.: MIT Press.
- Richards, Norvin. 2000. *Uttering Trees*. Linguistic Inquiry Monograph 56. The MIT Press, Cambridge, Massachusetts. MIT.
- Richards, Norvin. 2010. (Revised) *Uttering Trees*. MIT Press.
- Saito, Mamoru. 1985. *Some Asymmetries in Japanese and Their Theoretical Implications*. Doctoral dissertation. MIT, Cambridge, MA.
- Sohn, Keun Won. 1995. *Negative polarity items, scope and economy*. Doctoral dissertation. University of Connecticut, Storrs.

- Son, Gwangrak. 2015. The Nonstandard Paradigm of FNQ-Constructions as Topic Movement. *Linguistic Research*, 32(1):225-252.
- Sportiche, Dominique. 1988. A Theory of Floating Quantifiers and Its Corollaries for Constituent Structure. *Linguistic Inquiry*, 19(4):25-449.
- Takami, Ken-ich. 1998. *Nihongo no Suuryousi Yuuri Nituite [On Quantifier Float in Japanese]*. Gekkan Gengo 27(1), 86-95. Tokyo: Taishukan.
- Yip, Moira. 2002. *Tone*. Cambridge University Press.
- Yoshimoto, Kei, Masahiro Kobayashi, Jiroaki Nakamura and Yoshiki Mori. 2006. Processing of Information Structure and Floating Quantifiers in Japanese. In *New Frontiers in Artificial Intelligence*. ed. Takahi Washio, Akito Sakurai, Katsuto Nakajima, Hideaki Takeda, Satoshi Tojo, and Makoto Yokoo, 103-110. Springer.

# Detecting an Infant's Developmental Reactions in Reviews on Picture Books

Hiroshi Uehara<sup>†‡</sup> Mizuho Baba<sup>†</sup> Takehito Utsuro<sup>†</sup>

<sup>†</sup>Graduate School of Systems and Information Engineering, University of Tsukuba,  
Tsukuba, 305-8573, JAPAN

<sup>‡</sup>Corporate Sales and Marketing Division, NTT DOCOMO, INC.,  
Tokyo, 100-6150, JAPAN

## Abstract

We extract the book reviews on picture books written on the Web site specialized in picture books, and found that those reviews reflect infants' behavioral expressions as well as their parents' reading activities in detail. Analysis of the reviews reveals that infants' reactions written on the reviews are coincident with the findings of developmental psychology concerning infants' behaviors. In order to examine how the stimuli of picture books induces varieties of infants' reactions, this paper proposes to detect an infant's developmental reactions in reviews on picture books and shows effectiveness of the proposed method through experimental evaluation.

## 1 Introduction

Generally, educational books focus on a specific subject to be learned such as science, sociology, etc. Picture books are exceptions, in terms of their efficiency for infants' cognitive developments (Pardeck, 1986) without any intention on specific educational subject with their style of expressions, i.e., funny stories and pictures. Additionally, picture books are outstanding in that those who read them are separated from those who perceive them. Readers are parents or child care persons who make book talks for infants who do not have sufficient literacy yet. Infants perceive and interpret incoming stimuli of the book talks and the pictures.

According to the research in the developmental psychology, infants are found to express variety of cognitive reactions to the external stimuli in accordance with their developmental stage. If picture

books work as those kinds of stimuli, infants might express the cognitive reactions when the stimuli of picture books are perceived. Furthermore, this tendency might be amplified, because infants are free from understanding the printing letters of picture books.

In order to examine how the stimuli of picture books induces varieties of infants' reactions, we take an approach of applying a text mining technique to a large amount of the reviews on picture books written by their parents or the childcare persons. More specifically, this paper proposes to detect an infant's developmental reactions in reviews on picture books and shows effectiveness of the proposed method through experimental evaluation. This paper is the first attempt to solve the task of detecting an infant's developmental reactions in reviews on picture books.

## 2 The Web Site specialized in Picture Books

To analyze the infants' reactions, text data of reviews on picture books are collected from EhonNavi<sup>1</sup>, the web site specialized in picture books. EhonNavi provides with the information concerning picture books such as publishers, authors, outlines as well as a large amount of reviews written by the parents or child care persons, where the numbers of the titles of the picture books included in EhonNavi amount to about 55,600. The number of the reviews amount to approximately 290,000 as of January 2015 (shown in Table 1). Other than EhonNavi, popular Web sites

<sup>1</sup><http://www.ehonnavi>



Figure 1: An Example of a Review of "The Giant Turnip"

Table 1: Overview of EhonNavi

(a) Principal Information

Start date of the service	Number of titles	Number of unique users per month	Number of members	Number of reviews
Apr. 2002	55,600	1,055,000	343,000	289,000

(b) Distribution of the Numbers of Reviews according to Infants' Age

Age of infants	0	1	2	3	4	5
Number of reviews	7,272	13,450	22,448	25,795	21,573	18,143

Table 2: Categorization of Descriptions in Reviews

Categories		Explanation	Frequency in 345 reviews of 16 titles
Reviewers' reactions	impressions / critiques	Reviewers' impressions and / or critiques on the picture books	177
	retrospection in their ages of infants	Reviewers' retrospective descriptions reflecting their own reactions when they were in their infants' ages	11
	performance of reading	Performance such as gestures and change of voice tones for attracting the infants' attentions when reading	33
	expectation of infants' reactions	Reviewers' expectations and concerns about how the picture book affect to their infants	177
Infants' reactions		Infants' reactions to reviewers' reading of the picture books	276
Description of the story		Description of the scenes, stories, and the characters of the picture books	147

with a large amount of book reviews include Amazon<sup>2</sup> and Booklog<sup>3</sup>. Out of them, EhonNavi has a unique characteristics in that its reviews tend to be elaborated, reflecting the reactions of those who make book talks as well as those who perceive them. Additionally, it is also the EhonNavi's characteristics that the age of the infant is attached to each review. All these characteristics are preferable for our work aiming at detecting the infants' reactions in accordance with their developmental stages. Therefore, we employ the reviews of EhonNavi for the analysis of this paper.

### 3 Categorization of Descriptions in Reviews

Figure 1 shows an example of the review of EhonNavi. As shown in the figure, the header of each review includes the age of the infant to whom the reviewer reads the picture book. As described above, reviews of EhonNavi include descriptions of book talkers' reactions, mixed with infants' reactions. Since reviewers are book talkers in all the cases, infants' reactions described in reviews are those observed by reviewers.

<sup>2</sup><http://www.amazon.co.jp>

<sup>3</sup><http://booklog.jp>

In order to categorize descriptions in reviews, we randomly picked up 345 reviews from 16 titles of picture books and manually classified descriptions in those reviews<sup>4</sup>. Table 2 shows the result of categorizing descriptions in reviews. Those descriptions are roughly categorized into reviewers' reactions, infants' reactions, and descriptions of the story. Reviewers' reactions are further sub-categorized as shown in the table<sup>5</sup>. In order to further sub-categorize infants' reactions, we refer to studies of developmental psychology. In those developmental psychology literatures, they present categories of infants' cognitive developments in accordance with their ages. Next section introduces those categories of infants' cognitive developments and analyze the reviews based on them.

<sup>4</sup>The first author of the paper worked on manually categorizing descriptions in reviews.

<sup>5</sup>Note that, since each review may include not only one type of reviewers' reactions but also other type of reviewers' reactions, or both of reviewers' and infants' reactions, etc., sum of the frequencies in 345 reviews of 16 titles is more than 345.

Table 3: Infants’ Reactions based on the Theory of Developmental Psychology and Typical Expressions

Characteristics of developmental reactions	Explanations and examples	Typical expressions	
		ID	expression
Reactions to visual stimuli	Showing an interest in the pictures especially the ones of foods. / Enjoy to find something in the pictures that are familiar to the infants.	1.	gaze at / stare hard / listen hard
Physical expressions mixed with verbal expressions	Pointing fingers and making gestures in case the infants are not able to express verbally. / Reaching for the things on the picture book as if they were the real things.	2.	point fingers
Pretend play	An example: If the infant is asked to hand something to his or her parents, he or she pretends to hand it to them even though it does not exist.	3.	pretend
Imitate	Imitating various things such as the persons, things, and the events surrounding the infant.	4.	imitate
Supposition	Finding common characteristics between real things and what are supposed to be.	5.	suppose
Reactions to repeating the same rhythm	Reacting to onomatopoeic words. / The infant repeats the onomatopoeic words because of their rhythmical sounds, though he or she does not understand what they mean.	6.	onomatopoeic words
Game of make-believe	Reproducing the story of the picture book based on such activities that the infant imagines himself/herself to be in the place in the picture book.	7.	game of make-believe
Interests in the relationship or the causality	Indicating intellectual curiosity by asking “why” frequently. / An example: “Does Papa read the newspaper because he works? Does Mama cook the dinner because she is a housewife?”	8.	“?” (question mark)
Empathy for the story	Emotionally being involved in the world depicted by the picture book. / An example: “If I could enter into the picture book, I would save the cat.”	9.	enter into
		10.	empathy

Table 4: Number of Analyzed Reviews per age (ages from 0 to 5)

age of infants	0	1	2	3	4	5	total
number of reviews	1,491	3,150	4,306	4,062	3,203	2,033	18,245

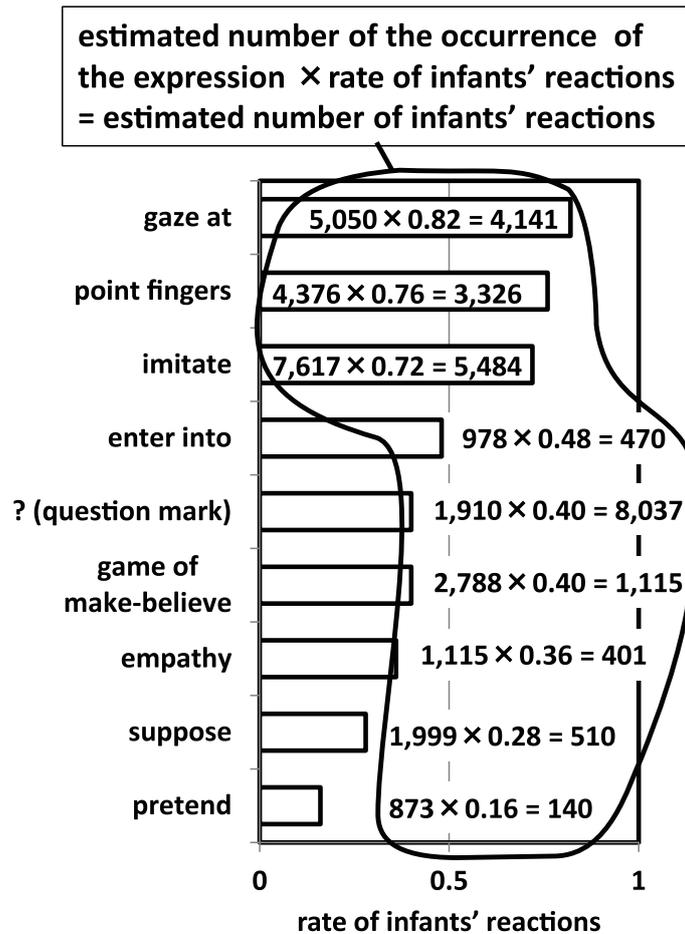


Figure 2: Estimating the Numbers of Infants' Developmental Reactions

#### 4 Categorizing Infants' Reactions based on Developmental Psychology

According to the theory of developmental psychology, infants express age specific reactions to incoming stimuli. We collect such infants' reactions that are specific to ages ranging from 0 to 3 from publications or papers concerning developmental psychology (Sully, 2000; Piaget, 1962; Leslie, 1987; Walker-Andrews and Kahana-Kalman, 1999) and list them in Table 3. In this table, we list those 10 types of reactions in the order of from those observed in the early age 0 to those observed in the later age 3. This result indicates that infants at their very early stage of ages tend to react automatically with their physical expression, such as pointing the fingers, or grasping gestures, meanwhile, those at their later stage of ages tend to react consecutively

expressing their intention, such as game of make-believe, or asking why, though some reactions are common over multiple ages.

Finally, we manually examine those randomly picked up 345 reviews from 16 titles of picture books examined in the previous section and collect typical expressions representing each of the 10 types of infants' reactions listed in Table 3. Collected expressions are shown on the right hand side column of Table 3.

#### 5 Detecting an Infant's Developmental Reactions in Reviews

The underlying motivation of this paper is to develop a system for recommending picture books which might induce expected infants' reactions specified by the users. Considering this motivation, this section examines whether it is possible to detect an in-

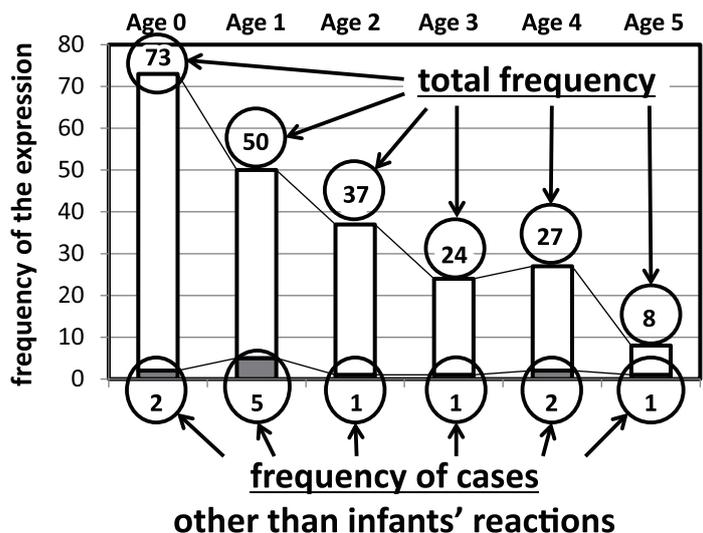


Figure 3: Frequency Distribution of Infants’ Reactions of the Expression per Age: “gaze at / stare hard / listen hard”

fant’s developmental reactions in reviews on picture books.

In order to select sample reviews for the analysis, we first collect titles of picture books which have sufficient number of reviews. Here, we rank picture books in descending order of the number of reviews and select the topmost 99 titles, where the total number of the reviews of those 99 titles amount to 27,661. Out of the total 27,661 reviews, we analyze those with infants of ages from 0 to 5 years old, which amount to 18,245 reviews, as shown in Table 4. Table 4 also shows the numbers of the analyzed reviews per age.

**5.1 Estimating the Numbers of Infants’ Developmental Reactions**

For most of the 10 types infants’ reactions as well as their typical expressions listed in Table 3, Figure 2 shows the rate of infants’ reactions within the occurrence of each expression as well as the estimated numbers of infants’ developmental reactions. For each of the 10 types of infants’ reactions, the rate of infants’ reactions within the occurrence of its typical expressions is measured by collecting the latest 20 reviews which include one of those typical expressions and then by manually examining whether each of their occurrences actually represents an infant’s developmental reaction or not.

Also, those estimated numbers of infants’ devel-

opmental reactions are calculated by measuring the number of the occurrence of the typical expressions listed in Table 3, and then by multiplying it by the rate of infants’ reactions within the occurrence of each expression.

As can be seen from this result, the rates of infants’ reactions are relatively low. In the next section, we propose to detect an infant’s developmental reactions with collocational expressions so that we can improve the rate of infants’ reactions.

**5.2 Detecting an Infant’s Developmental Reactions in Reviews with Collocational Expressions**

As typical expressions which represent infants’ developmental reactions and are suitable for the analysis of this paper, out of the 10 types infants’ reactions as well as their typical expressions listed in Table 3, we select “gaze at / stare hard / listen hard”, “imitate”, and “game of make-believe”. According to the studies in developmental psychology (Sully, 2000; Piaget, 1962; Leslie, 1987; Walker-Andrews and Kahana-Kalman, 1999), the infants’ reaction “gaze at / stare hard / listen hard” is mostly observed around the age of 1, “imitate” around that of 2, and “game of make-believe” around that of 3. Then, in order to detect an infant’s developmental reactions in reviews on picture books, we propose to collect collocations of each of those three expres-

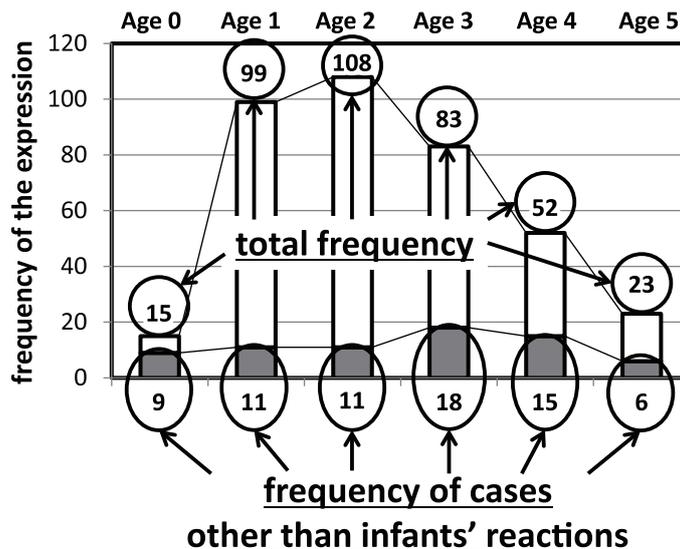


Figure 4: Frequency Distribution of Infants’ Reactions of the Expression per Age: “imitate”

sions as below and to detect an infant’s developmental reactions represented by those collocational expressions.

- For “gaze at / stare hard / listen hard”, we further add pronunciation variation of “gaze at / stare hard / listen hard” as well as expressions which are “gaze at / stare hard / listen hard” concatenated with the object “the picture book”.
- For “imitate”, we collect “imitate to eat”, “imitate and”, and “imitation of”.
- For “game of make-believe”, we collect “make-believe play” and “train games”.

For each of the three expressions “gaze at / stare hard / listen hard”, “imitate”, and “game of make-believe”, Figure 3 to Figure 5 show frequency distribution of infants’ reactions per age. In these figures, total frequencies of those with collocational expressions as well as frequencies of cases other than infants’ reactions are shown. Those frequencies are counted by manually judging several hundreds matched expressions. From these results, we measure rates of correctly detecting infants’ developmental reactions, which are 94% (“gaze at / stare hard / listen hard”), 77% (“imitate”), and 70% (“game of make-believe”). Thus, it is quite possi-

ble to detect an infant’s developmental reactions in reviews on picture books with fairly high precision.

Moreover, out of all the occurrences of each of the expressions “gaze at / stare hard / listen hard”, “imitate”, and “game of make-believe”, we examine how many of them are actually covered by the collected collocational expressions. We found that those with the collected collocational expressions cover about 50% (“gaze at / stare hard / listen hard”), 60% (“imitate”), and 66% (“game of make-believe”) of their occurrences. Thus, those collected collocational expressions cover fairly large amount of occurrences. Finally, as clearly shown in this result, expressions appearing in reviews of EhonNavi represent infants’ reactions in the way coincident with respective age specific reactions asserted by developmental psychology.

## 6 Conclusion

In order to examine how the stimuli of picture books induces varieties of infants’ reactions, this paper proposed to detect an infant’s developmental reactions in reviews on picture books and showed effectiveness of the proposed method through experimental evaluation. Future work includes developing a framework of recommending picture books which accepts the age of an infant and an expected developmental reaction as its input, and as its output, gives a list of picture books that are ranked according to

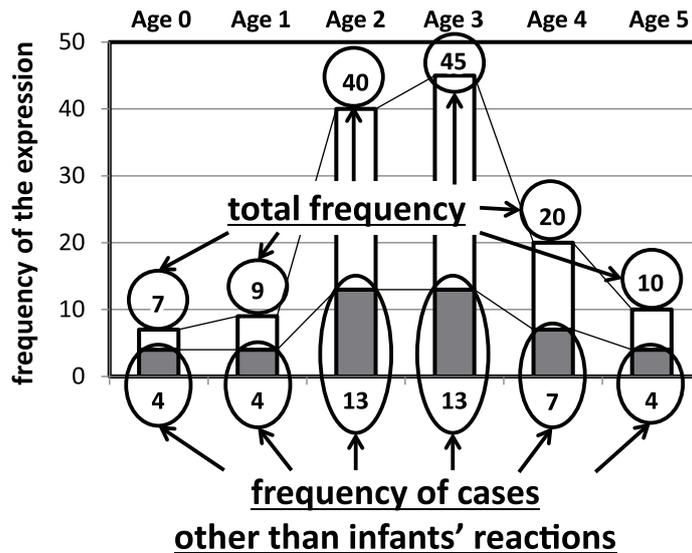


Figure 5: Frequency Distribution of Infants' Reactions of the Expression per Age: "game of make-believe"

the degree of expected developmental reactions by infants.

**References**

A. M. Leslie. 1987. Pretense and representation: The origins of theory of mind. *Psychological Review*, 94(4):412-426.

J. Pardeck. 1986,. *Books for Early Childhood: A Developmental Perspective*. Greenwood Pub Group.

J. Piaget. 1962,. *Play, Dreams, and Imitation in Childhood*. W W Norton & Co Inc.

J. Sully. 2000. *Studies of Childhood*. Free Association Books.

A. S. Walker-Andrews and R. Kahana-Kalman. 1999. The understanding of pretence across the second year of life. *British Journal of Developmental Psychology*, 17(4):523-546.

# Semi-automatic Filtering of Translation Errors in Triangle Corpus

**Sung-Kwon Choi, Jong-Hun Shin and Young-Gil Kim**

Natural Language Processing Research Section

Electronics and Telecommunications Research Institute, Daejeon, Korea

{choisk, jhsin82, kimyk}@etri.re.kr

## Abstract

We are developing a multilingual machine translation system to provide foreign tourists with a multilingual speech translation service in the Winter Olympic Games that will be held in Korea in 2018. For a knowledge learning to make the multilingual expansibility possible, we needed large bilingual corpus. In Korea there were a lot of Korean-English bilingual corpus, but Korean-French bilingual corpus and Korean-Spanish bilingual corpus lacked absolutely. Korean-English-French and Korean-English-Spanish triangle corpus were constructed by crowdsourcing translation using the existing large Korean-English corpus. But we found a lot of translation errors from the triangle corpora. This paper aims at filtering of translation errors in large triangle corpus constructed by crowdsourcing translation to reduce the translation loss of triangle corpus with English as a pivot language. Experiment shows that our method improves +0.34 BLEU points over the baseline system.

## 1 Introduction

Triangle corpus is the corpus ‘source language-pivot language-target language (hereafter, L1-Lp-L2)’ where a source language (hereafter, L1) is translated into a pivot language (hereafter, Lp) and then the pivot language is translated into a target language (hereafter, L2). One of methods building large triangle corpus with English as a pivot language is a crowdsourcing translation. The crowdsourcing translation means a distributed model of translation that uses contributors instead of, or combined with, professional translators. In environment of the crowdsourcing translation it is

possible to build a lot of bilingual corpora that are both time and cost-effective. In particular, if there is large bilingual corpus of L1-English, we can produce fast translation result of L1-L2 via the crowdsourcing translation of English-L2. Although there is such advantage of crowdsourcing translation, there is also its drawback that translation errors and inconsistency can arise because a large pool of people is going to generate input of differing quality. That is, a translation loss can be produced between L1-English and English-L2.

This paper aims at semi-automatic filtering of translation errors in large triangle corpus constructed by crowdsourcing translation to reduce the translation loss that can occur in crowdsourcing translation of corpus with English as a pivot language. The remainder of this paper is organized as follows. Section 2 presents the related work. In Section 3, we describe large English-French and English-Spanish bilingual corpus constructed by crowdsourcing translation using English as a target language of large Korean-English corpus. Translation errors in the Korean-English-French triangle corpus are manually analyzed by a human translator. In Section 4, we describe how to filter the translation errors caused from the crowdsourcing translation. Section 5 presents the experimental setup and the results.

## 2 Related Work

There were very little researches to improve the procedural translation loss of L1-English-L2 triangle corpus. Instead, there have been numerous researches in machine translation (hereafter, MT)

using L1-English-L2 corpus as a training set. Such researches can be classified into three methods.

- **Transfer Method:** the transfer method (Utiyama and Isahara, 2007; Costa-jussà et al., 2011) connects a source-pivot MT system and a pivot-target MT system. The source-pivot MT system translates a source sentence into the pivot language, and the pivot-target MT system translates the pivot sentence into the target sentence. The problem with the transfer method is that the time cost is doubled and the translation error of the source-pivot translation system will be transferred to the pivot-target translation because it needs to decode twice.
- **Synthetic Method:** the synthetic method creates a synthetic source-target corpus by: (1) translate the pivot part in source-pivot corpus into target language with a pivot-target model; (2) translate the pivot part in pivot-target corpus into source language with a pivot-source model; (3) combine the source sentences with translated target sentences or/and combine the target sentences with translated source sentences (Wu and Wang, 2009). The problem with the synthetic method is that it is difficult to build a high quality translation system with a corpus created by a machine translation system.
- **Triangulation Method:** the triangulation method obtains source-target phrase table by merging source-pivot and pivot-target phrase table entries with identical pivot language phrases and multiplying corresponding posterior probabilities (Cohn and Lapata, 2007). According to an Arabic-Chinese experiment of Chen et al.(2008), BLEU(Papineni et. al. 2002) of statistical machine translation (hereafter, SMT) based on the triangulation method was better than that of SMT based on L1-L2. The problem of this approach is that the probability space of the source-target phrase pairs is non-uniformity due to the mismatching of the pivot phrase. To resolve this disadvantage, Zuh et al.(2014) proposed the approach to calculate the co-

occurrence count of source-pivot and pivot-target phrase pairs.

Despite these three methods, there were still little researches in checking what kind of translation loss the L1-English-L2 triangle corpus has. Furthermore, there were little evaluation about corpus which was constructed by crowdsourcing translation. In this point, this paper aims at semi-automatic filtering of translation errors of large L1-English-L2 triangle corpus constructed by crowdsourcing translation to reduce the translation loss.

### 3 Human Analysis of Translation Errors in Crowdsourcing Translation

We are developing a multilingual MT system including Korean, English, Chinese, Japanese, French, Spanish, German, and Russian to provide foreign tourists with a multilingual speech translation service in the Winter Olympic Games that will be held in Korea in 2018. The multilingual MT system is characterized as follows:

- **Controllability:** makes high-quality translation possible through manual correction of knowledge errors by users and obtains the effect of the aforesaid customization.
- **Common transfer:** makes the addition of new languages easy because many languages share a format of transfer such as universal dependency annotation for multilingual parsing (McDonald et al., 2013)
- **Knowledge learning:** makes multilingual expansibility and/or domain customization possible because the translation knowledge is automatically learned from training data.

Our multilingual MT system considers in particular a multilingual expansibility as important. For a knowledge learning to make the multilingual expansibility possible, we needed large bilingual corpus. In Korea there were a lot of Korean-English (hereafter, K-E) bilingual corpus, but either Korean-French (hereafter, K-F) bilingual corpus or Korean-Spanish (hereafter, K-S) bilingual corpus lacked absolutely. It was very expensive to construct the K-F and K-S bilingual

corpus by professional translators. We had to think about constructing K-F and K-S corpus by crowdsourcing translation using the existing large K-E bilingual corpus. That is, English of K-E bilingual corpus became a source language and was translated into French and Spanish respectively. Crowdsourcing translation was conducted by Flitto in Korea, a global crowdsourcing translation platform like Amazon’s Mechanical Turk (Callison-Burch and Dredze, 2010). K-F and K-S bilingual corpus constructed by crowdsourcing translation were as follows.

	# of sentences	Build-up period
K-E corpus	779,382	
E-F corpus	100,000	1 month
E-S corpus	200,000	1 month

Table 1: E-F and E-S corpus constructed by crowdsourcing translation using K-E corpus

200,000 of English sentences whose word length is in  $3 < \# < 23$  became candidate sentences for K-F corpus and K-E corpus. Table 1 indicates that E-S corpus had 100,000 more sentences than E-F corpus because Flitto, crowdsourcing translation company held more English-Spanish translators than English-French translators.

To check translation quality in crowdsourcing translation, we extracted randomly 500 K-E-F sentences from 100,000 K-E-F sentences and conducted a human analysis of translation errors. The translation error analysis was based on the translation accuracy, which means conveying correctly the meaning of source sentence to the meaning of target sentence. K-E and E-F sentences were analyzed respectively. Types of translation errors include not only existing error types in machine translation (Fishel et al., 2012; Popovic et al., 2011) but also new error types such as ill-formed source sentence, ungrammatical generation and misunderstanding of situation. The result of analysis was as follows.

Types of translation errors	# of K-E sentences	# of E-F sentences	# of K-F sentences
Missing words-	2	3	5

noun			
Missing words - pronoun	0	2	2
Missing words - negation	0	1	1
Incorrect words - verb	1	46	47
Incorrect words - noun	6	29	32
Incorrect words - relative pronoun	0	1	1
Incorrect words - article	0	1	1
Incorrect words - adverb	0	5	5
Incorrect words - preposition	0	6	6
Incorrect words - auxiliary verb	0	1	1
Incorrect words - adjective	0	1	1
ungrammatical generation - tense	0	5	5
ungrammatical generation - grammar	4	6	10
misunderstanding of situation	16	1	17
ill-formed source sentence	9	12	15
Total	38	120	149
500	7.6%	24.0%	29.8%

Table 2: Translation error analysis in 500 K-E-F sample sentences

In Table 2, the second column indicates the number of translation errors in K-E bilingual corpus constructed by professional translators and shows that 38 of 500 sentences have translation errors. The third column presents the number of translation errors in E-F sentences that were translated from English sentences of K-E bilingual corpus to French sentences by crowdsourcing and shows that 120 of 500 sentences have translation errors. The error analysis of the second and third column was separately conducted. In the fourth column it turns out that the K-F bilingual corpus as a combination between K-E translation and E-F translation has 149 sentences with translation

errors which run to 29.8% of 500 sentences. Through Table 2, we can know that the translation errors in L1-L2 corpus of L1-English-L2 corpus come from a combination of both the translation errors of L1-English and the translation errors of English-L2. The following examples show such cases.

Example 1: Error of K-F translation due to the error of K-E human translation

Korean source sentence: “배수의 진을 쳤다.” (“I make a last-ditch fight.”)

K-E Human translation: “I was between the devil and the deep blue sea.”

E-F Crowdsourcing translation: “J’étais en plein dilemme.” (“I was in a dilemma.”)

Example 2: Error of K-F translation due to the error of E-F crowdsourcing translation

Korean source sentence: “아무 때라도 좋습니다.” (“Anytime is okay.”)

K-E Human translation: “Anytime.”

E-F Crowdsourcing translation: “Je vous en prie.” (“You’re welcome”)

Example 1 shows a K-F translation error due to the error ‘incorrect words –noun’ of K-E human translation. The Korean source sentence “배수의 진을 쳤다” that means “I make a last-ditch fight” was wrongly translated into the French sentence “J’étais en plein dilemme” that means “I was in a dilemma” because the Korean source sentence was wrongly translated into the English sentence “I was between the devil and the deep blue sea”. Example 2 presents the error of K-F translation due to the error ‘misunderstanding of situation’ of E-F crowdsourcing translation. The Korean source sentence “아무 때라도 좋습니다” that means ‘Anytime is okay’ was wrongly translated into the French sentence “Je vous en prie” that means “You are welcome” because the English sentence “Any time” was wrongly translated into the French sentence “Je vous en prie” that means “You’re welcome”.

#### 4 Assuming Distances in Triangle Corpus

In this section, we show a series of effort to find the sentence pairs including translation errors in crowdsourcing translation. Our goal is to find sentences which have content words that are

semantically wrong. A general approach to realize this goal will be to use a bilingual dictionary. But it is difficult to build the bilingual dictionary. Besides, we need the part-of-speech tagger to align the words between source language and target language. To use a comparable corpus for under-resourced languages was also difficult. From this reason, we tried to measure the semantic distance by using L1-Lp-L2 without using a comparable corpus.

A vectorial text representation which is called a distributed word representation is a method to capture semantic and syntactic similarity of words in a monolingual sentence. (Bengio et al., 2003; Mikolov et al., 2013) Previous works on a distributed word representation have been concentrated on a monolingual corpus or have been approach to learn the linguistic regularities which are generalized across languages. (Klementiev et al., 2012; Lauly et al., 2014; Hermann and Blunsom, 2014a, 2014b) Such existing studies are based on the following idea: similar semantic and syntactic properties will be embedded nearby in the embedded vector space. We denote the representation result as a bilingual word embedding. Such representations have been used to achieve an excellent performance on word sense disambiguation, cross-lingual information retrieval, and word alignments. In this paper, we also use the characteristics of bilingual word embedding.

#### 4.1 Motivations and System Structures to Find Translation Errors in Crowdsourcing Translation

When we construct the triangle corpus with English as a pivot, the following problems arise: 1) the translation errors appear due to missing words and grammatical errors, and 2) the meaning difference between L1-Lp sentences and Lp-L2 sentences affects the meaning difference between L1-L2 sentences. In case we implement a SMT system using such triangle corpus, the corpus including translation errors can cause the word alignment mismatching and have a bad influence on the translation quality of the SMT system. To resolve such problems, we tried to measure a sentence distance of L1-Lp-L2 and a sentence distance of L1-L2 respectively to find the semantic or syntactic similarity, since we thought that the similarity might be a clue of translation errors such

as semantic alternation, misprints and missing words. So, we used the bilingual distributed word representation.

Before measuring the sentence distance, the bilingual word embedding was constructed. Given the multilingual parallel corpus consisting of  $n$  language pairs including a specific source language,  $n(n+1)/2$  of embedding should be produced. We conducted the word segmentation in Korean. In this paper we measured the distance between embeddings to extract the sentences L1-Lp-L2 that are beyond the threshold.

## 4.2 Calculating a Sentence Distance of L1-Lp-L2

The distributed word representation presents as a set of fixed-column real valued weights, and each weight can be assumed as a dimension. So we can handle a word of a sentence as a vector point in a hyperspace which can be calculated with a vector distance function.

Suppose we are given set of word pairs and their associated vector representation  $\{x_i, y_i\}_{i=1}^n$ , where  $x_i \in \mathbb{R}^{d1}$  is the vector representation of word  $i$  in the source language, and  $y_i \in \mathbb{R}^{d2}$  is the vector representation of word in target language. We calculate similarity for each word vector in a sentence, by the following  $n$ -dimensional cosine distance function:

$$d_1(x, y) = 1 - \cos\theta = 1 - \frac{\sum_{i=1}^n x_i y_i}{\sqrt{\sum_{i=1}^n x_i^2} \sqrt{\sum_{i=1}^n y_i^2}} \quad (1)$$

And Euclidean distance function considered as alternative to measure sentence distance:

$$d_2(x, y) = \frac{\sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2 + \dots + (x_n - y_n)^2}}{\sqrt{\sum_{i=1}^n (x_i - y_i)^2}} \quad (2)$$

We applied cosine distance functions to set of words in a source-pivot sentence pair and a source-target sentence pair. By looking for a minimum distance to each of the words constituting the given sentence, it will be assist to find improper used vocabulary or absence of core keywords. So, a distance of each sentence is defined as equation (3):

$$SDist(S^{d1}, S^{d2}) = \frac{\sum_{i=1}^n \sum_{j=1}^m \text{argmin}(d_1(a_i, b_j))}{n} \quad (3)$$

where  $a_i$  is  $i$ -th word of a source sentence  $S^{d1}$ , and  $b_j$  is  $j$ -th word of a target sentence  $S^{d2}$ , relatively( $a_i \in S^{d1}, b_j \in S^{d2}$ ). After calculating distance of L1-Lp and Lp-L2 sentence, we need to calculate a complete distance with following equation. given a source language sentence  $S^{dS}$ , a pivot language sentence  $S^{dP}$ , and a target language sentence  $S^{dT}$ , equation (4) is a final ‘averaged’ distance of  $S^{dS} - S^{dT}$ :

$$\text{AvgSentDist}(S^{dS}, S^{dP}, S^{dT}) = \frac{1}{\sqrt{(SDist(S^{dS}, S^{dP}) + SDist(S^{dP}, S^{dT}) - SDist(S^{dS}, S^{dT}))^2}} \quad (4)$$

We wanted to find whether there is any correlation between the distance and the translation quality, even if we measure the distance of content words in L1-Lp-L2 through the above equation. It was because we had to establish a criterion about how long distance was wrongly translated to find the sentence pairs with translation errors. In our experiment, human translators decided heuristically whether the sentence pairs have a similar meaning in the statistical distribution of a calculated distance.

## 5 Experimental Result

### 5.1 Data and parameters

To verify the performance of the proposed methods, we used Korean-English-French corpus consisting of 100,000 parallel sentences. We tokenized and lowercased the English and French sentences, using some useful corpus preprocessing scripts in cdec-decoder. (Dyer et al., 2010) And for Korean we used in-house Korean morphological analyzer to get word tokens instead of using a monotonic whitespace tokenizer. To learn the bilingual word embedding, we used BICVM (Hermann and Blunsom, 2014a). Models were trained for up to 50 iterations. We set a dimensionality of word embedding size to  $D=128$  as a default parameter and set the number of noise elements to 200. The adaptive gradient method (Duchi et al., 2011) was used to update weights of the models.

### 5.2 Filtering Experiment by Sentence Distance

We now present the calculated distance results by using our methodology. Test sentences were 300, which are in low order of calculated distances. The error analysis was as follows:

Incorrect Translations	Correct Translations	Total
114	186	300

Table 3: Error analysis of 300 sample sentences

If a Korean sentence was ambiguous, but a French sentence was correctly translated from its English sentence, we considered the Korean-French sentence pair as a correctly translated sentence pair. We analyzed the sentences that were incorrectly translated. They were 114 sentences which consisted of error types such as missing target word, irrelevant translation, incomplete sentences, and meaning change. Detailed error types were as follows:

Missing target word	Irrelevant translation	Incomplete sentences	Meaning changes	Total
14	8	16	76	114

Table 4: Error types of incorrect translation

Most errors of “missing target word” were error type “missing noun word” (11 of 14 sentences, 78%). The meaning changes due to the literal translation occurred in French sentences with narrowish meaning via the ambiguous predicates in English sentences (35 of 76 sentences, 46%). The examples of sentences with incorrect translation are shown in below table:

1	KO	습관성 턱 관절 탈골이에요.
	EN	He is tendency temporomandibular dislocation.
	FR	Je ne comprends pas cette phrase, désolé.
2	KO	그 은행이 계좌를 개설하면 고작 금반지를 나눠준대.
	EN	The bank only gives away foil when you open an account.
	FR	La banque ne donne que.
3	KO	정리를 해 주세요.

	EN	Please take care of it.
	FR	S'il vous plaît occupez - vous en.
4	KO	저는 개를 좋아합니다.
	FR	J'aime les chiens.
5	KO	더 보고 싶으신 건 없나요?
	FR	Avec ceci?

Table 5: Examples of Translation Errors

In the case of first sentence example, the French sentence “Je ne comprends pas cette phrase, désolé” means “I cannot understand that phrase, I’m sorry...”. We guess that a crowdsourcing participant translated the French sentence so because he/she did not understand the meaning of a medical term ‘temporomandibular dislocation’. In the second example, French sentence that means “the bank only gives away” was not completed unlike Korean and English sentence. In the third example, Korean sentence means “Please clean up” or “Please arrange it”. But it was incorrectly translated into “Take care of it” in English and “S’il vous plaît occupez - vous en” in French that means “Please take care of you”. And the fourth Korean sentence was correctly translated into both English sentence and French sentence in the point of view of common speech (or slang). The last example is considered as a bad translation because the French sentence means “with this?” literally, even if it has same meaning as “is there anything else?” in French cultural area. Like this, translated sentences are dependent on cultural differences and slang/common speeches.

### 5.3 Verifying Experiment of Sentence Distance using Phrase-based SMT

To compare a performance of a filtered Korean-English-French corpus with a performance of an original Korean-English-French corpus, we trained a phrase-based SMT (Koehn et al., 2007). 90,000 sentences were a training set and the remaining 10,000 sentences were an evaluation set in order to train a SMT model. To make a filtered corpus, we removed the farthest distance of 1,000 sentences from the calculated sentence distance list, which would be assumed the incorrectly translated sentences. The sentences removed from training set were 919 sentences. So, sentences to train a filtered SMT model became 89,081. 87 sentences

were removed from the evaluation set, so we used 9,913 sentences for a performance evaluation. The evaluation metric of SMT model was BLEU (Papineni et al., 2002). Along with this evaluation set, we conducted an additional automatic evaluation using in-house Korean-French corpus which contains 3,000 parallel sentences with 1 reference. This evaluation set has same tourist/dialog domains as crowdsourcing translation corpus. Total number of Korean words were 12,284 and a sentence consisted of 4 words in average, while total number of French words were 20,346 and a sentence consisted of 6 words in average. The evaluation results are illustrated with below table:

	BLEU (Original)	BLEU (Filtered)
10k samples(pivot)	8.45	8.44
9.9k samples(pivot)	8.46	8.47
3k evalset(pivot)	14.13	14.47

Table 6: Original (=Non-filtered) / Filtered BLEU evaluation score result. 10k samples and 9.9k samples denote an evaluation corpus size, which is non-filtered original and filtered evaluation set respectively. And 3k evalset denotes our in-house Korean-French BLEU evaluation set.

In table 6, the ‘pivot’ denotes the transfer method (Wu and Wang, 2007), that is, Korean-English SMT results were used to get the translation results of the English-French SMT system. Despite of the simplicity of proposed method, the amount of the total training corpus was decreased, but we could see a slight performance improvement. From the above results, we could discover that removing the sentences which have a weak semantic similarity is helpful for improving translation corpus quality.

## 6 Conclusion

The crowdsourcing translation is an excellent method to reduce the translation cost and the translation period to construct large bilingual corpus. In case the corpus by the crowdsourcing translation is very large, the assessment of translation quality about the corpus should depend on the random sampling. Such random sampling could not resolve the translation loss caused by crowdsourcing translation.

This paper aimed at no random sampling, but the total crowdsourcing translation to be examined. Through word distance and sentence distance, we could extract high-quality translations of L1-L2 without translation loss from total crowdsourcing translation of L1-Lp-L2. Furthermore, our approach has the advantage to make efficient management of high quality multilingual corpus possible because it can reduce a translation loss due to triangulation translation and intensify L1-Lp-L2 due to a combination among languages.

## Acknowledgments

This work was supported by the ICT R&D program of MSIP/IITP. [10041807, Development of Original Software Technology for Automatic Speech Translation with Performance 90% for Tour/International Event focused on Multilingual Expansibility and based on Knowledge Learning]

## References

- Bengio, Y., Ducharme, R., Vincent, P., and Janvin, C. (2003) A neural probabilistic language model. The Journal of Machine Learning Research, 3, 1137-1155.
- Callison-Burch, C. and Dredze, M. (2010) Creating Speech and Language Data With Amazon’s Mechanical Turk. In Proceedings NAACL.
- Chen, Y., Eisele, A., and Kay, M. (2008) Improving Statistical Machine Translation Efficiency by Triangulation. In Proceedings of the Sixth International Language Resources and Evaluation, 2875-2880.
- Cohn, T. and Lapata, M. (2007). Machine Translation by Triangulation: Make Effective Use of Multi-Parallel Corpora. In Proceedings of 45th Annual Meeting of the Association for Computational Linguistics, 828-735.
- Costa-jussà, M.R., Henríquez, C., and Banchs, R.E. (2011). Enhancing Scarce-Resource Language Translation through Pivot Combinations. In Proceedings of the 5th International Joint Conference on Natural Language Processing, 1361-1365.
- Duchi, J., Hazan, E., and Singer, Y. (2011) Adaptive subgradient methods for online learning and stochastic optimization. Journal of Machine Learning Research, 12, 2121-2159.
- Dyer, C., Lopez, A., Ganitkevitch, J., Weese, J., Ture, F., Blunsom, P., Setiawan, H., Eidelman, V., and Resnik, P. (2010) A decoder, alignment, and learning

- framework for finite-state and context-free translation models. In *Proceedings of ACL System Demonstrations*, 7-12.
- Fishel, M., Bojar, O., and Popović, M. (2012) Terra: a Collection of Translation Error-Annotated Corpora. In *Proceedings of Language Resources and Evaluation Conference*, 7-14
- Hermann, K. M., and Blunsom, P. (2014a) The Role of Syntax in Vector Space Models of Compositional Semantics. In *Association for Computational Linguistics*, 894-904.
- Hermann, K. M., and Blunsom, P. (2014b) Multilingual models for compositional distributed semantics. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, 58 - 68.
- Klementiev, A., Titov, I., and Bhattarai, B. (2012). Inducing crosslingual distributed representations of words. In *Proceedings of COLING*, 1459-1474.
- Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., and Herbst, E. (2007) Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th annual meeting of the ACL on interactive poster and demonstration sessions*, 177-180.
- Laully, S., Boulanger, A., and Larochelle, H. (2014) Learning multilingual word representations using a bag-of-words autoencoder. *arXiv preprint arXiv:1401.1803*.
- McDonald, R., Nivre, J., Quirnbach-Brundage, Y., and Goldberg, Y. (2013) Universal Dependency Annotation for Multilingual Parsing. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*, 92-97.
- Mikolov, T., Sutskever, I., Chen, K., Corrado, G., and Dean, J. (2013) Distributed Representations of Words and Phrases and their Compositionality. In *Proceedings of NIPS*.
- Papineni, K., Roukos, S., Ward, T., and Zhu, W. J. (2002) BLEU: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on association for computational linguistics*, 311-318.
- Popović, M. and Ney, N. (2011) Towards Automatic Error Analysis of Machine Translation Output. *Computational Linguistics*. Vol.37, Number 4, 657-688.
- Utiyama, M. and Isahara, H. (2007). A Comparison of Pivot Methods for Phrase-Based Statistical Machine Translation. In *Proceedings of Human Language Technology: the Conference of the North American Chapter of the Association for Computational Linguistics*, 484-491.
- Wu, H., & Wang, H. (2007). Pivot language approach for phrase-based statistical machine translation. *Machine Translation*, 21(3), 165-181.
- Wu, H. and Wang, H. (2009). Revisiting Pivot Language Approach for Machine Translation. In *Proceedings of the 47th Annual Meeting of the Association for Computational Linguistics and the 4th IJCNLP of the AFNLP*, 154-162.
- Zuh, X., He, Z., Wu, H., Zhu, C., Wang, H., and Zhao, T. (2014) Improving Pivot-based Statistical Machine Translation by Pivoting the Co-occurrence Count of Phrase Pairs. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 1665–1675.

# Cross-language Projection of Dependency Trees for Tree-to-tree Machine Translation

Yu Shen<sup>1</sup>, Chenhui Chu<sup>2</sup>, Fabien Cromieres<sup>2</sup>, Sadao Kurohashi<sup>1</sup>

<sup>1</sup>Graduate School of Informatics, Kyoto University

<sup>2</sup>Japan Science and Technology Agency

shen-yu@nlp.ist.i.kyoto-u.ac.jp, (chu, fabien)@pa.jst.jp, kuro@i.kyoto-u.ac.jp

## Abstract

Syntax-based machine translation (MT) is an attractive approach for introducing additional linguistic knowledge in corpus-based MT. Previous studies have shown that tree-to-string and string-to-tree translation models perform better than tree-to-tree translation models since tree-to-tree models require *two* high quality parsers on the source as well as the target language side. In practice, high quality parsers for both languages are difficult to obtain and thus limit the translation quality. In this paper, we explore a method to transfer parse trees from the language side which has a high quality parser to the side which has a low quality parser to obtain transferred parse trees. We then combine the transferred parse trees with the original low quality parse trees. In our tree-to-tree MT experiments we have observed that the new combined trees lead to better performance in terms of BLEU score compared to when the original low quality trees and the transferred trees are used separately.

## 1 Introduction

Depending on whether or not monolingual parsing is utilized, there are about 4 types of machine translation (MT) methods. string-to-string (Koehn et al., 2007; Chiang, 2005), string-to-tree (Galley et al., 2006; Shen et al., 2008), tree-to-string (Liu et al., 2006; Quirk et al., 2005; Mi and Huang, 2008), and tree-to-tree (Zhang et al., 2008; Richardson et al., 2014).

Though the tree-to-tree system that employs syntactic analysis for both source and target sides seems

to be the best intuitively, in practice, two good quality parsers are difficult to acquire which affects the translation quality which is sensitive to the differences in syntax annotation. In many cases, one parser is of a much higher quality than the other since one of the languages is easier to parse and has a well annotated treebank. In case of Japanese to Chinese translation, Japanese is easier to parse than Chinese and Japanese parsers typically make fewer mistakes compared to Chinese parsers.

In this paper, we explore a method which relies on using parallel text for transferring syntactic knowledge from a high quality (HQ) parser to a low quality (LQ) parser using alignment information (Ganchev et al., 2009; Hwa et al., 2005). Henceforth we shall refer to Japanese as HQ or HQ side, indicating that it is the language which has a high quality parser. Conversely Chinese will be referred to as the LQ or LQ side since the Chinese parser is of a relatively lower quality and makes a number of parsing mistakes. One advantage is that the transferred parse information will possibly be more similar to the other side's parse. This will also reduce the parsing error on the LQ side and unify the syntactic annotation on both sides.

This idea has been proposed before, but not much has been done in the case of dependency-based tree-to-tree SMT system, which is the setting of this paper. Furthermore, this method results in two types of trees on the LQ side: The original LQ tree and the tree transferred from the HQ side. These two trees have their individual strengths: The transferred tree could be more precise compared to the original LQ one in theory, but it is much more sensitive to

alignment errors or bad parallel sentences (not direct translation). To address these problems, previous studies simply apply language dependent rules to the transferred trees, for example in English *have* and *be* must have an object modifier. In this paper we consider combining these two trees and get improved results. We show in our experiments that combining the LQ-parsed trees with the transferred trees yield better translation results rather than only using them individually.

## 2 Related Work

### 2.1 Syntax Transfer for Non-MT Task

There are many previous works describing methods to improve the performance of NLP tasks for a resource poor language by using a related resource rich language (mainly English). Amongst these the ones which employ methods which transfer information perform better than unsupervised methods. (Das and Petrov, 2011) describe an approach for inducing unsupervised part-of-speech tags for languages that have no labeled training data. (Jiang et al., 2010) show a transfer strategy to construct a constituency parser. (Ganchev et al., 2009) present a partial, approximate transfer through linear expectation constraints to project only parts of the parse trees to the low resource language side.

However, improving monolingual parsing accuracy does not directly lead to higher MT performance, as it does not address the annotation criteria difference problem.

### 2.2 Syntax Transfer for MT Task

For MT tasks, most transfer based works assume that the source side has a poor tree-bank, a bad quality parser and/or little training data (parallel corpus to be precise). For phrase-based models, (Goto et al., 2015) proposed a cross transfer pre-ordering model which employs a target-language syntactic parser without requiring a source language parser. For tree based models, most works focus on the fact that they have no source side parse tree and create a parse tree with transfer. (Jiang et al., 2010) showed that a transferred constituent tree parser leads to results that are comparable with those obtained using a supervised tree parser. (Hwa et al., 2005) worked on transferring the results of an English parser to

a resource poor language and applied post-transfer transformations like *An aspectual marker should modify the verb to its left*.

These previous works typically assume that only one out of the two languages has a parser. In practice however, the resource poor language has a parser, whose quality is worse than that of a resource rich language’s parser. In this work, we consider such a setting. If we combine the transferred parse tree with the original parse tree, the performance should improve.

## 3 The Difficulties of Tree-to-Tree Approaches

A tree-to-tree model is the most natural in the MT scenario because it respects syntax. However for tree-to-tree translation models, we need both a good target-side parser and a good source-side parser. Even if the parsers are of high quality, we may have problems due to different syntax annotation criteria. Our main objective is that we want to improve the translation quality a dependency-based tree-to-tree system such as kyotoEBMT (Richardson et al., 2014). The differences between the tree-to-string model and tree-to-tree models are shown in Fig 1 and Fig 2.

In Fig 2, it can be seen that the target side parsing error affects tree-to-tree system’s search space. It shows that a relatively low quality parser limits the system. Unlike the tree-to-string approach, the tree to tree approach is affected by the parsing error on both sides. Another problem is shown in Fig 3, where the coordination relation ‘wo he ta (my and his)’ in the Chinese parse tree (bottom side) is annotated as ‘siblings’, but in Japanese side (top side) these three words ‘watashi oyobi karen (my and his) have a parent-child relation. Even if both of them are correct in their own tree-banks, tree-to-tree decoder won’t match a sibling relation pair with a parent-child relation pair as is explained in Fig 2.

A transfer based method should solve these problems, because it makes LQ trees more similar to the HQ trees. Transferring HQ side syntax relation not only fixes the LQ side parsing error, but also unifies the syntax annotation criterion.

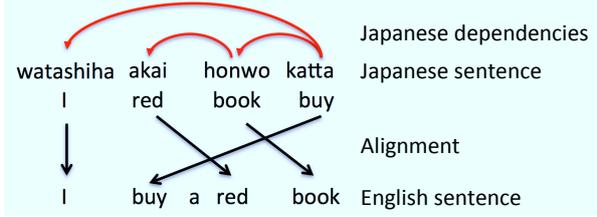


Figure 1: For tree-to-string system, it searches the case *watashi katta* (I buy) *akai honwo* (red book) because there are dependency pairs in source parse tree and does not search the case *watashi akai* (I red) because it is not a pair in source parse tree

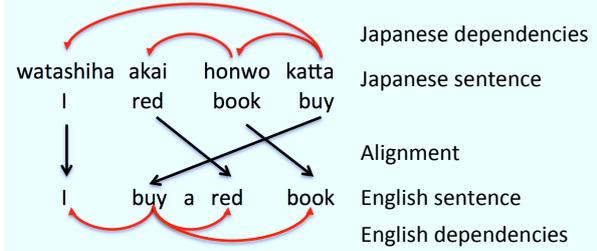


Figure 2: For tree-to-tree system, it searches the case *watashi katta* (I buy) because it is a dependency pair in source parse tree and target parse tree. It does not search the case *watashi akai* (I red) because it is not a dependency pair in source parse tree neither in target parse tree. Unlike Fig.1, it does not search *akai honwo* (red book) because although it is a dependency pair in source side, it is disconnected in target side

## 4 Transfer of Syntactic Dependencies

### 4.1 Overview and Notation

This section gives an overview of our approach. In the description below we use dependency tree structure. To represent dependency we will use the following notation  $tree = \{(i, j), \dots\}$ . It means that the word in position  $j$  is the parent of the word in position  $i$ ; we use  $(i, -1)$  to represent that  $i$  is the root. To represent alignment we will use the following notation  $a = \{i-j, \dots\}$ . It means that the word in the source side position  $i$  is aligned to the word in the target side position  $j$ .

By this we mean that, we first transfer the entire high quality (HQ) dependency tree  $Trees$  to low quality (LQ) side which replaces the original LQ tree  $Tree_T^{old}$  to a transferred tree  $Tree_T^{new}$  (Section 4.2). For each word  $w_i$  in LQ side sentence, parent of  $w_i$  is denoted by  $w_{i_{newp}}$  after transfer and  $w_{i_{oldp}}$  before transfer. This direct mapping method always transfers source side syntax structure to the target side by alignment regardless of the possibil-

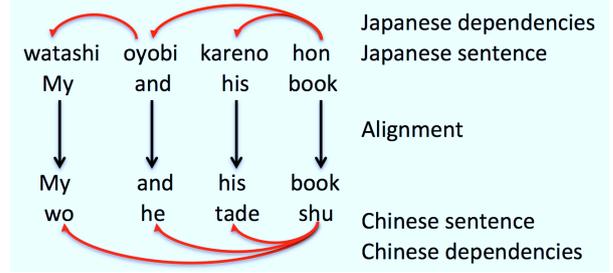


Figure 3: An example of different syntax annotation

ity of alignment error or different expression. In practice, if we test  $Tree_T^{new}$  in our tree-to-tree system, we get about a 2 point reduction in BLEU. As a result, the second step contains a backstep which makes some  $w_{i_{newp}}$  back to  $w_{i_{oldp}}$  for keeping the sentence projective. Notice that the original monolingual dependency tree is always projective (Most parsers give projective results). The worst case is to let all  $w_{i_{newp}}$  point back to  $w_{i_{oldp}}$ , i.e. to keep the dependency tree structure unmodified (Section 5.2). In HQ-LQ (input-side has high-quality parser) MT task, for each training parallel sentence, creating a combined tree for LQ side is enough. However in the LQ-HQ (the input-side has Low-Quality parser) MT task, transferring training data is not enough. For the input LQ side sentence, it makes no sense to use the original monolingual LQ parse tree. Thus, for the third step, we re-train a new LQ side parser using the combined data (Section 6).

### 4.2 Transfer Dependencies

Intuitively, mapping a high accuracy syntax parser to a low accuracy syntax parser will lead to better performance and the success of this approach depends on the quality of word alignment on a parallel corpus. This Direct Mapping (DM) can be formalized as below:

Given a sentence pair  $(S, T)$  where  $S = s_1 s_2 \dots s_n$  is a sentence of HQ parse side and  $T = t_1 t_2 \dots t_n$  is a sentence of LQ parse side, a dependency tree for  $S$  denoted as  $Trees = \{(s_i, s_j) \dots\}$  which has been mentioned before. The new LQ parse tree  $Tree_T^{new}$  is transferred from HQ parse tree  $Trees$  as follows.

- one to one case: If  $s_i$  aligns to a unique  $t_j$ ,  $s_x$  aligns to a unique  $t_y$ , and  $(s_i, s_x) \in trees$ , push  $(t_j, t_y)$  into  $tree_T^{new}$ .
- one to many case : If  $s_i$  aligns to  $t_x \dots t_y$ , then

take one of them as representative, a tree based alignment should let  $t_x..t_y$  be a treelet. We take the root of  $t_x..t_y$  as representative and then perform the same steps as in the one-to-one case. For the node  $t_z$  other than representative  $t_r$ , we simply push  $(t_z, t_r)$ .

- many to one case : If  $s_i..s_j$  aligns to  $t_x$ , like the one to many case, take one of them as a representative. A tree based alignment should let  $s_i..s_j$  be a treelet, take the root of  $s_i..s_j$  as the representative and then perform the same steps as in the one-to-one case.
- many to many case: Reduce this to one-to-many and many-to-one cases, i.e. both side select a representative and then perform the same steps as in the one-to-one case.
- unaligned case (HQ side): If  $s_i$  is an unaligned word, just treat it as non-existent and link two sides of  $s_i$ . More specifically, if  $s_i$  is not aligned,  $(s_i, s_j) \in Trees$ . and  $(s_k, s_i) \in Trees$ , push  $(s_k, s_j)$  into  $Trees$ .
- unaligned case (LQ side): If  $t_i$  is an unaligned word, just push  $(t_i, t_i + 1)$  or  $(t_i, t_i - 1)$  into  $Tree_T^{new}$ .

Direct Mapping (DM) gives us a simple way of obtaining dependency tree parsing and there are many works that investigate these kinds of mapping and show that they work well. We, however, still want to test the efficacy of direct mapping. We train a new parser based on transferred data and show that it leads to lower parsing score (Section 7.1).<sup>1</sup> This shows that the DM approach won't directly improve the parsing accuracy.

## 5 Post Transfer Transformations

### 5.1 Error Analysis

We check the difference between the correct annotation and transferred trees. There are three differences.

- different annotation criterion on both sides (false error): Because the labels were designed

<sup>1</sup>The transferred tree which has been tested in Section 7.1 is the combined tree after the method in Section 5.2 is applied

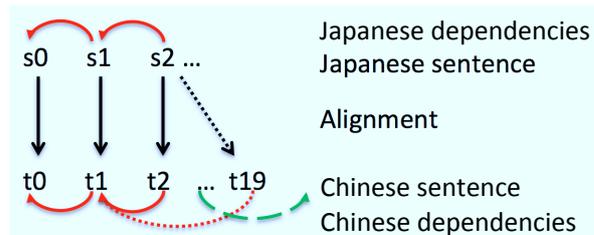


Figure 4: This is an example of alignment error where the solid line is the correct alignment and the dashed line is the wrong alignment. It shows that incorrect alignments lead to parsing error by DM

for a monolingual scenario, there is always a difference in the annotation criteria. For example: *Shanghai Industrial Technology school* on Ja and Zh. The gold standard data looks like:  $Trees = \{(0, 1)(1, 2)(2, 3)(3, -1)\}$   $Tree_T^{old} = \{(0, 3)(1, 3)(2, 3)(3, -1)\}$ , assume alignment is  $a = \{0-0, 1-1, 2-2, 3-3\}$ . By Direct Mapping,  $Tree_T^{new} = \{(0, 1)(1, 2)(2, 3)(3, -1)\}$  is the same as the one on the Ja side but completely wrong when comparing it to the gold standard tree. The reason is Ja side tends to let words of compound nouns be child-parent pair and Zh tends to let words of compound nouns be sibling. We call it false error, though the dependency score will decrease by comparing it with gold standard data, it actually helps tree-based translation system retrieve this Chinese compound noun better than before because its structure now is much more similar to the Japanese side.

- alignment error (true error): The direct mapping method is highly dependent on alignment. If alignment is incorrect, direct mapping which uses this erroneous alignment gives a wrong dependency result. See Fig.4 for an example. This is quite critical; originally a tree-to-tree MT system might reject this wrong alignment by the distance on tree with feature or criterion like a child-parent pair in source side should translate to a child-parent pair in target side. We now force a child-parent pair to align to a child-parent pair using DM which will prevent the tree distance criterion from influencing the translation accuracy.
- different expression (true error): Sometimes

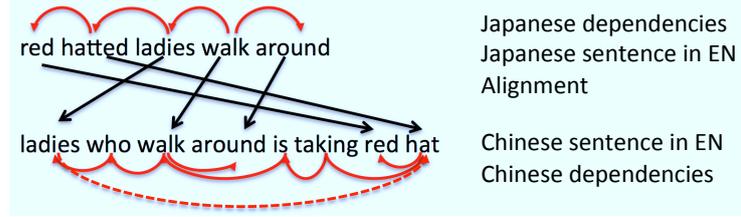


Figure 5: This is an example of a different expression where again the solid line is the correct parse and the dashed line is the wrong parse caused by DM.

the translation is not a direct translation, for example a Ja sentence with the meaning ‘ladies wearing red hats walk around’ will sometimes be translated to Zh with meaning ‘ladies who walk around are taking red hats’. Even though the alignment is perfect, the direct mapping method makes the Zh words for ‘ladies’, ‘red’ and ‘hat’ take on child-parent relations which, ofcourse, is a parsing error as shown in Fig.5.

### 5.2 Keep the Tree Projective

Error analysis for true and false errors allowed us to revise our approach to incorporate a criterion *projectivity* that can distinguish a good mapping between the true and false errors discussed above. *projectivity* is a property of parse tree which means there shouldn’t be any crosses in the tree structure. For example:  $Tree_T^{new} = \{(0, 2)(1, 3)(2, 3)(3, -1)\}$  is not projective. To be more precise, for two child-parent pairs  $(a, b), (c, d)$ , we denote the interval  $[a, b]$  (if  $a > b$ ),  $swap(a, b)$  as  $span(a, b)$ . Here statement  $(span(a, b) \text{ cross } span(c, d)) \text{ equal } (c \in [a, b]) \text{ and } (d \notin [a, b])$ . Notice that the root node is denoted as  $(c, -1)$ ,  $(d \notin [a, b])$  is always true. Many alignment errors cases can be detected by the property of projectivity. For example, in Fig.4, a parent-child pair  $(19, 2)$  created by erroneous alignment looks very strange and has a high probability of having crosses with other dependencies like the green dependency in the figure.

Many different expression cases can also be detected by the property of projectivity. Review the example Fig.5 in the previous section. Here child-parent pair  $(7, 0) \text{ hat, ladies}$  is a long distance dependency relation which crosses with other dependency pairs like  $(5, -1) \text{ talking root}$ .

Although annotation criterion are different, both of them are reasonable for showing dependency relations. Thus a mapping one to another still

keeps the tree projective. Consider the example mentioned above, *Shanghai Industrial Technology school* on JP and ZH. Original monolingual Chinese dependency tree  $Tree_T^{old} = \{(0, 3), (1, 3), (2, 3)(3, -1)\}$  and transferred tree  $Tree_T^{new} = \{(0, 1), (1, 2), (2, 3), (3, -1)\}$  are BOTH projective.

Thus keeping the tree projective prevents many projection errors. Notice that  $Tree_T^{old}$  is always projective. We introduce a back search method which makes some words relations in  $Tree_T^{new}$  back to relations in  $Tree_T^{old}$ . The pseudo-code shown below:

---

#### Algorithm 1 Back Searching

---

```

1:  $len \leftarrow Tree_T^{new}.length$ 
2: for  $i = 1..len$  do
3:   for  $j = 1..len$  do
4:     find  $x$  s.t.  $(t_i, t_x) \in Tree_T^{new}$ 
5:     find  $y$  s.t.  $(t_j, t_y) \in Tree_T^{new}$ 
6:     if  $span(i, x)$  cross with  $span(j, y)$  then
7:       find  $z$  s.t.  $(t_i, t_z) \in Tree_T^{old}$ 
8:        $(t_i, t_x) \leftarrow (t_i, t_z)$ 
9:       find  $z$  s.t.  $(t_j, t_z) \in Tree_T^{old}$ 
10:       $(t_j, t_y) \leftarrow (t_j, t_z)$ 
11:     if a loop is created then undo

```

---

For a non-projective part, which means it creates cross in the dependency tree, We trust the original monolingual dependency tree and for projective part we retain the direct mapping result. More precisely, with  $span(i, x)$  cross with  $span(j, y)$ , we find  $z$  that  $(t_i, t_z) \in Tree_T^{old}$  and substitute  $(t_i, t_x)$  with  $(t_i, t_z)$ . This operation actually moves  $t_i$ ’s parent back to the state before mapping. Doing the same thing to  $t_j$ , we find  $z$  that  $(t_j, t_z) \in Tree_T^{old}$  and substitute  $(t_j, t_y)$  with  $(t_j, t_z)$ .

For non-aligned word  $t_i$ , our strategy is a little different. In the direct mapping process, we did not change the non-aligned word’s dependency because

we didn't have any information to decide its parent. When we encounter non-projectivity on  $t_i$ , we simply change  $t_i$ 's parent to  $t_j$  which ensures projectivity. Changing  $t_i$ 's parent to  $t_j$  solves the cross between  $span(i, x)$  and  $span(j, y)$ . In any case we try to retain its original parent as much as possible.

Sometimes back searching like above leads to loops in the tree. This happens when other non-projective parts should be solved before this part. We check whether a loop exists in the tree after each back operation to  $t_i, t_j$  and undo the modification if so. We run several iterations of the procedure mentioned above till it reaches the worst case which means reverse  $Tree_T^{new}$  is the same as  $Tree_T^{old}$ .

It is not so simple to test the effectiveness of *projectivity* on tasks other than MT. We manually check the percentage of errors our method has solved by manually evaluating 50 sentences (Section 7.2).

## 6 Re-train a New LQ Side Parser

Using the word alignments and original monolingual dependency trees, we successfully create combined trees using the parallel training corpus. As we have mentioned before, this is enough for the HQ-LQ MT task but still a bit not enough for LQ-HQ MT task. For a LQ side sentence as an input, it makes no sense to use the original monolingual LQ side parser which now has a different annotation criterion since it uses the combined trees in training corpus. Considering we have abandoned the original dependency tree, we now regard the combined trees as 'golden data' and train a new model with these 'golden data' using a LQ side parser. After that, for an input sentence, we utilize the new parser rather than the original one.

## 7 Experiment

### 7.1 Parsing Accuracy

We conducted a Chinese parsing experiment on scientific domain. The Chinese parser used in our experiment is the SKP parser (Shen et al., 2012).<sup>2</sup> As the baseline parser, we trained SKP with the Penn Chinese treebank version 5 (CTB5) containing 18k sentences on news domain, and a in-house treebank which contains about 10k sentences in scien-

<sup>2</sup><https://bitbucket.org/msmoshen/skp-beta>

tific domain, with default parameters. The new combined parser that we proposed used the training data obtained from the ASPEC Ja-Zh parallel corpus,<sup>3</sup> containing 670k sentences. We used a Japanese parser KNP (Kawahara and Kurohashi, 2006)<sup>4</sup> and the baseline SKP to automatically parse these sentences. We then created combined Chinese dependency trees.<sup>5</sup> Finally, we trained a new parser using these combined Chinese dependency trees with the same parameters.

As test data we used an additional 1k sentences from our in-house treebank. Table 1 shows the results of these two parsers.

Parser	UAS	Root-Accuracy
Baseline	0.7433	0.6950
Combined	0.5890	0.6140

Table 1: Parsing accuracy

Because we deliberately ignored the annotation criterion problem and labelled the dependencies of non-aligned words quite freely, the decrease of accuracy is not surprising (according to us).

### 7.2 Projectivity for Solving True Errors

In addition, we evaluated our new parse trees in another way. There are two true errors for the DM approach which have been discussed in Section 5.1, some of them should be rejected by the property of *projectivity*, some of them could not. We randomly selected 50 sentences to check how many errors of these two error types have been solved.

Case	Solve rate
Alignment error	90%
Different expression	55%

Table 2: The percentage of alignment error and different expression which has been solved by projectivity

The results are shown in Table 2. We can see that projectivity is an effective way of addressing the alignment error problem, but the *different expression*

<sup>3</sup><http://lotus.kuee.kyoto-u.ac.jp/ASPEC/>

<sup>4</sup><http://nlp.ist.i.kyoto-u.ac.jp/EN/index.php?KNP>

<sup>5</sup>We only obtained 419k combined trees, as the rest were unchanged sentences.



Figure 6: An example of different expression can't be detected by projectivity, solid line is the correct parse and the dashed line is the wrong parse caused by DM.

problem is much harder to detect. Fig 6 shows an example of the different expression problem. A possible solution is make some language based rules like *the word before possessive's parent is the word after it*.

### 7.3 Translation

We conducted experiments for Japanese-to-Chinese (Ja-Zh) and Chinese-to-Japanese (Zh-Ja) translation. For both tasks, we used the ASPEC Ja-Zh parallel corpus as training data. We used 2,090 and 2,107 additional sentence pairs for tuning and testing, respectively. In our experiments, we compared the MT performance of our proposed projection method with the baseline parser. We used a tree-to-tree system KyotoEBMT for our experiments (Richardson et al., 2014).<sup>6</sup> To parse the Chinese and Japanese sentences, we again used SKP and KNP, respectively.

In order to test our combined tree approach, we substituted the original SKP parsing results to combined parse trees. We also tested the direct mapping method which simply transfers the Japanese parse trees to the Chinese side. In Ja-Zh task, we trained a new Zh parser by using the combined parse trees.

System	Ja-Zh	Zh-Ja
Moses	27.25	33.94
KyotoEBMT	29.08	35.10
Direct map	27.28	33.23
combined map	29.89*	35.59*

Table 3: BLEU scores for ASPEC JA-ZH and ZH-JA. (\* denotes that the result is significantly better than 'KyotoEBMT' at  $p < 0.05$ )

Table 3 shows the results. For reference, we also show the MT performance of the phrase based

<sup>6</sup><http://nlp.ist.i.kyoto-u.ac.jp/EN/index.php?KyotoEBMT>

system Moses, which is based on the open-source GIZA++/Moses pipeline (Koehn et al., 2007). We also conduct significance tests which were performed using the bootstrap resampling method proposed by Koehn (2004). The Direct mapping method which does not use the Chinese parser decreases the MT performance. The combined tree method works pretty well. In the Ja-Zh direction, it gets a 0.8 BLEU score improvement and 0.5 in the Zh-Ja direction. The biggest problem for the Zh-Ja direction is that we have to feed automatic data to SKP for training a new parser. SKP is designed to work with manually annotated training data (gold data set) but not automatically generated training data. The best evidence is that, if we parse a training sentence with this parser, the result is quite different with the original training data. For Ja-Zh direction, it is quite straightforward, we just combine SKP and KNP parsing results to create a Ja like Zh dependency tree for each training sentence. Combined tree increased the BLEU score but decreased parsing accuracy since parsing accuracy is tested on monolingual test data. Linguistic parse tree structures are not the most appropriate for a tree-to-tree MT system.

## 8 Conclusion and Future Work

In this paper, we have proposed a method to use both source side and target side parse trees to create a combined parse tree which improves the BLEU score on a tree-to-tree MT system. Transferring parsing information from a relatively high accuracy parser on the source language side to the target language side with a relatively low accuracy parser constrained by *projectivity* performs well. It not only fixes the parsing error of the low accuracy side, but also addresses the problem of different annotation criteria on both sides.

We used automatically created, combined parse tree to train a new parser by using an existing parser. It is not very appropriate because parsers are designed to train on gold standard data and not automatic data. Though it leads to some positive results, we think that the quality could be further improved. Instead of re-parsing the parser, we could also save the different parts in the combined parse trees, and apply them to the input sentences. This could be much more efficient and logical than re-parsing but a bit difficult when considering issues such as how to save and how to apply them to the input sentences.

## References

- David Chiang. 2005. A hierarchical phrase-based model for statistical machine translation. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, pages 263–270, Ann Arbor, Michigan, June. Association for Computational Linguistics.
- Dipanjan Das and Slav Petrov. 2011. Unsupervised part-of-speech tagging with bilingual graph-based projections. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 600–609, Portland, Oregon, USA, June. Association for Computational Linguistics.
- Michel Galley, Jonathan Graehl, Kevin Knight, Daniel Marcu, Steve DeNeefe, Wei Wang, and Ignacio Thayer. 2006. Scalable inference and training of context-rich syntactic translation models. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 961–968, Sydney, Australia, July. Association for Computational Linguistics.
- Kuzman Ganchev, Jennifer Gillenwater, and Ben Taskar. 2009. Dependency grammar induction via bitext projection constraints. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pages 369–377, Suntec, Singapore, August. Association for Computational Linguistics.
- Isao Goto, Masao Utiyama, Eiichiro Sumita, and Sadao Kurohashi. 2015. Preordering using a target-language parser via cross-language syntactic projection for statistical machine translation. *ACM Trans. Asian Low-Resour. Lang. Inf. Process.*, 14(3):13:1–13:23, June.
- Rebecca Hwa, Philip Resnik, Amy Weinberg, Clara Cabezas, and Okan Kolak. 2005. Bootstrapping parsers via syntactic projection across parallel texts. *Natural language engineering*, 11(03):311–325.
- Wenbin Jiang, Yajuan Lv, Yang Liu, and Qun Liu. 2010. Effective constituent projection across languages. In *Coling 2010: Posters*, pages 516–524, Beijing, China, August. Coling 2010 Organizing Committee.
- Daisuke Kawahara and Sadao Kurohashi. 2006. A fully-lexicalized probabilistic model for japanese syntactic and case structure analysis. In *Proceedings of the Human Language Technology Conference of the NAACL, Main Conference*, pages 176–183, New York City, USA, June. Association for Computational Linguistics.
- Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondrej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics Companion Volume Proceedings of the Demo and Poster Sessions*, pages 177–180, Prague, Czech Republic, June. Association for Computational Linguistics.
- Yang (1) Liu, Qun Liu, and Shouxun Lin. 2006. Tree-to-string alignment template for statistical machine translation. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 609–616, Sydney, Australia, July. Association for Computational Linguistics.
- Haitao Mi and Liang Huang. 2008. Forest-based translation rule extraction. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, pages 206–214, Honolulu, Hawaii, October. Association for Computational Linguistics.
- Chris Quirk, Arul Menezes, and Colin Cherry. 2005. Dependency treelet translation: Syntactically informed phrasal SMT. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, pages 271–279, Ann Arbor, Michigan, June. Association for Computational Linguistics.
- John Richardson, Fabien Cromières, Toshiaki Nakazawa, and Sadao Kurohashi. 2014. Kyotoebmt: An example-based dependency-to-dependency translation framework. In *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 79–84, Baltimore, Maryland, June. Association for Computational Linguistics.
- Libin Shen, Jinxi Xu, and Ralph Weischedel. 2008. A new string-to-dependency machine translation algorithm with a target dependency language model. In *Proceedings of ACL-08: HLT*, pages 577–585, Colum-

- bus, Ohio, June. Association for Computational Linguistics.
- Mo Shen, Daisuke Kawahara, and Sadao Kurohashi. 2012. A reranking approach for dependency parsing with variable-sized subtree features. In *Proceedings of the 26th Pacific Asia Conference on Language, Information, and Computation*, pages 308–317, Bali, Indonesia, November. Faculty of Computer Science, Universitas Indonesia.
- Min Zhang, Hongfei Jiang, Aiti Aw, Haizhou Li, Chew Lim Tan, and Sheng Li. 2008. A tree sequence alignment-based tree-to-tree translation model. In *Proceedings of ACL-08: HLT*, pages 559–567, Columbus, Ohio, June. Association for Computational Linguistics.

# Realignment from Finer-grained Alignment to Coarser-grained Alignment to Enhance Mongolian-Chinese SMT

**Jing Wu Hongxu Hou Congjiao Xie**

College of Computer Science

Inner Mongolia University

Hohhot, 010021, China

cshhx@imu.edu.cn

## Abstract

The conventional Mongolian-Chinese statistical machine translation (SMT) model uses Mongolian words and Chinese words to practice the system. However, data sparsity, complex Mongolian morphology and Chinese word segmentation (CWS) errors lead to alignment errors and ambiguities. Some other works use finer-grained Mongolian stems and Chinese characters, which suffer from information loss when inducting translation rules. To tackle this, we proposed a method of using finer-grained Mongolian stems and Chinese characters for word alignment, but coarser-grained Mongolian words and Chinese words for translation rule induction (TRI) and decoding. We presented a heuristic technique to transform Chinese character-based alignment to word-based alignment. Experimentally, our method outperformed the baselines: fully finer-grained and fully coarser-grained, in terms of alignment quality and translation performance.

## 1 Introduction

Mongolian is an agglutinative language and has complex morphology. The current scale of Mongolian-Chinese parallel corpus is very small. These two reasons make data sparsity a very serious

problem in Mongolian-Chinese SMT. Using finer-grained Mongolian stems rather than Mongolian words can reveal the word semantics and alleviate data sparsity. On the other hand, CWS is a necessary process to separate Chinese words, because Chinese words are not naturally separated by space (Jiang et al., 2009). CWS can achieve high accuracy, but does not necessarily guarantee better performance of alignment (Chang et al., 2008; Zhang et al., 2008; Xiao et al., 2010). Besides, CWS also brings errors (Xiao et al., 2010). Using of finer-grained Chinese characters, which are separated without using of CWS, can avoid the CWS errors and alleviate data sparsity. However, coarser-grained basic units are proved perform better in translation rule induction (TRI). (Philipp Koehn et al., 2003).

So inspired by the work of (Xi et al., 2011; Xi et al., 2012), we proposed a method that uses different granularity respectively for alignment and TRI. We train a finer-grained alignment using Mongolian stems and Chinese characters. Afterwards, we realign it to Chinese words and Mongolian words alignment for the following TRI and decoding. We design a technique to convert finer-grained alignment to coarser-grained alignment. The conversion can be unambiguous after carefully processing the differences brought by Mongolian word lemmatization and CWS.

In the experiments, our method outperformed the baselines of fully finer-grained and fully coarser-grained, in terms of alignment quality and translation performance. The experiments indicate that using finer-grained basic units for alignment and

coarser-grained basic units for TRI performs better than other granularity combinations.

The rest of the paper is organized as follows: Section 2 explains how our method designed and how can it have good influence on alignment and translation. Section 3 demonstrates the realignment model and analyzes how it works for better alignment. Section 4 describes the evaluations. Section 5 is the conclusion.

## 2 Design of different Granularity Alignment

The conventional practice of SMT uses Mongolian and Chinese words in the process of word alignment and TRI (Brown et al., 1993). We proposed a method of using finer granularity for word alignment but coarser granularity for TRI to enhance the Mongolian-Chinese SMT system. The process of the method is depicted in Figure 1:

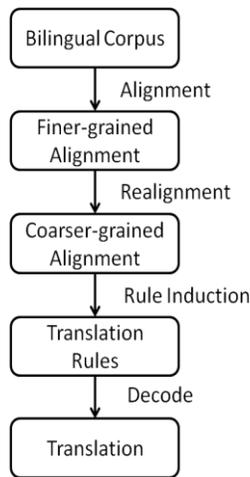


Figure 1. Process of the method

- (1) In the first step, we get the finer-grained alignment by using Mongolian stems and Chinese characters as basic units;
- (2) In the realignment procedure, we transform finer-grained alignment into coarser-grained alignment through a converting technique;
- (3) In the step of TRI and decoding, we use the coarser-grained alignment.

Mongolian words are formed by stems and suffixes (Hou et.al., 2000). For some examples: when a noun plays different constituents in sentence, like subject or object, the case suffixes added to it are different; a verb adds different inflectional suffixes

when it is under different tenses or followed by different nouns; a word has different forms (with the same word stem but different suffixes) when it is in different positions of the sentence. Therefore, Data sparsity is a very serious problem in Mongolian-Chinese SMT because of the complex Mongolian morphology and the small scale parallel corpus. Mongolian stems-based alignment can mitigate this problem, because Mongolian words in different forms but with the same semantic meaning will become one same stem after removing some suffixes. Besides, using Chinese characters for alignment can avoid the errors brought by CWS. Table 1 shows the token distribution of Mongolian words and Mongolian stems in corpus. We can see that the unique tokens in stem-based corpus reduce almost 10% than those in word-based corpus. Table 2 shows the frequency distribution of words and characters of Chinese corpus. The tokens whose frequency is no than 4 has a lower percentage in character-based corpus. We see that the unique tokens in character segment corpus are only one-third of those in word segment corpus. In the fined-grained Chinese corpus, the frequency of 77.88% tokens are equal to or more than 5, while the percentage of word tokens in coarser-grained Chinese corpus is only 38.74%. The above statistical data prove that coarser-grained word alignment suffers from more serious data sparsity than finer-grained word alignment.

	Word	Stem
Total Tokens	37140	29861
Unique Tokens	20859	14340
Percentage (%)	56.16	48.02

Table 1. Unique tokens of Mongolian word and stem

Frequency	Word (%)	Character (%)
1	31.25	9.34
2	14.91	5.85
3	8.84	3.47
4	6.26	3.46
5+	38.74	77.88

Table 2. Frequency distribution of Chinese word and character

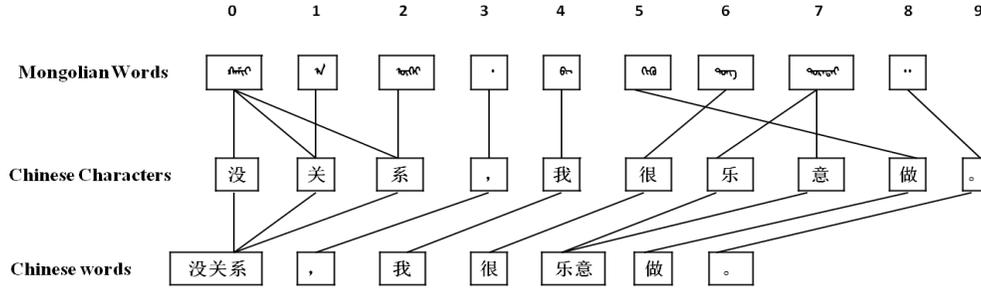


Figure 2. Realignment from finer-grained to coarser-grained

However, comparing to finer-grained tokens, coarser-grained tokens have more complete semantic information. State-of-the-art SMT models achieve excellent results by extracting phrases to induct the translation rules (Philipp Koehn et al., 2003). When the phrase-based translation models try to extract and score the phrases by getting lexical translation table, the probability of words to words can express more semantic information than stems to characters (Deng and Zhou, 2009). Moreover, when we use language model, the position information expressed by Mongolian word suffixes might be ignored by using Mongolian stems. Therefore, we still use coarser-grained units to induct the translation rules.

### 3 Realignment

The realignment from Mongolian stems to Mongolian words is an easy method of one-to-one mapping because there is no position changing. We build a heuristic model to describe the Chinese realignment. We set  $e$  and  $f$  as the source (Mongolian) and target (Chinese) sentence in finer-grained alignment. Given finer-grained source sentence (Mongolian)  $e'$  and target sentence (Chinese)  $f'$ , we can get the coarser-grained alignment  $a$  by the realignment model as equation (1):

$$P(a|e, f) = P(a_c|e', f')P(a|a_c) \quad (1)$$

In the model,  $a_c$  is the finer-grained alignment getting from  $e'$  and  $f'$ .  $P(a_c|e', f')$  is the alignment model used in our alignment training which can be given as log-linear model by (Och and Ney, 2005; Liu et al., 2005) as equation (2).

$$P(a_c|e', f') = \frac{\exp[\sum_{m=1}^M \lambda_m h_m(a_c, e', f')]}{\sum_{a_c} \exp[\sum_{m=1}^M \lambda_m h_m(a_c, e', f')]} \quad (2)$$

The conversion model  $P(a|a_c)$  can be modeled based on (Zhang, 2003) as equation (3):

$$P(a|a_c) = P(a, a_c)/P(a_c) \quad (3)$$

It is easy to understand that the transformation from a finer-grained sentence to its coarser-grained sentence is unambiguous. An example of conversion shows in figure 2, we can get the word alignment from Mongolian words to Chinese words by converting Chinese characters into Chinese words. “没关系” is a Chinese word which means “It does not matter”. It is composed of three characters “没”, “关” and “系”. The alignment from Mongolian words to Chinese characters “没”, “关” and “系” is “0-0, 0-1, 0-2, 1-1, 2-2”, the alignment from Chinese characters to Chinese word “没关系” is “0-0, 1-0, 2-0”, so the realignment from Mongolian words to Chinese word is “0-0, 1-0, 2-0”. Another example shows in figure 2 is the alignment from Mongolian word “ᠯᠡᠮᠠᠭᠠᠨ” to Chinese word “乐意”, which means “with pleasure”. Comparing with Chinese words, Chinese characters carry more uncertain meaning. “关” is a verb which means “close”, but when it is followed by “系”, which is also a verb and means “tie”, the meaning of “关系” is “relation” and it is a noun. So using Chinese characters as basic unit may induce more interference alignment options. However, the recall score gets higher when we apply Chinese characters to do the alignment. Because we find that when we get the word alignment by realigning

No.	0 1 2 3 4 5 6 7 8 9 10
Mongolian Word	ᠮᠣᠩᠭᠣᠯᠢᠨ ᠠᠨᠠᠭᠤᠯᠢᠰᠤᠨ
Chinese Word	飞机 着 陆 时 ， 请 把 座 椅 调 直 。
Coarser-grained Alignment	0-0 1-2 3-3 5-6 9-4 9-5 6-6 7-6 8-6 9-6 9-7 8-8 10-9
Realignment	0-0 1-2 2-2 3-2 4-1 4-2 5-2 5-3 7-5 6-6 8-6 8-7 8-8 9-6 10-9

Figure 3. Comparing coarser-grained alignment with realignment

from Chinese character-based alignment rather than by Chinese words directly, there are fewer invalid alignment options. We believe this problem is avoided by the feature of co-occurrence and distortion used by alignment models which explained in detail by (Xi et al., 2012).

Moreover, we find that our method can mitigate word alignment errors which caused by incorrect CWS result. As shows in figure 3, the correct segmentation should be “飞机 着陆时” but not “飞机 着陆时”. The correct alignment should make No.1, No.2, No.3, No.4 and No.5 of Mongolian words align to the No.1 and No.2 of Chinese words, but Chinese word-based alignment only gets the right alignment 1-2, but wrongly aligns No.3 of Mongolian word to the Chinese comma as showed in the fourth row of figure 3. In our method, we can get a more precise alignment result based on characters “着”, “陆” and “时”. The realignment based on a wrong word segmentation result will lead to a wrong word alignment inside the phrase “着 陆时”. However, as showed in the fifth row of the figure 3, we find that because of the better character-based alignment, the phrase “着 陆时” as a whole still can be realigned more precisely to its corresponding Mongolian phrase.

In conclusion, due to a more precise Chinese character-based alignment, our realignment based on Chinese word segmentation (even based on a wrong word segmentation result) can get a more precise word alignment result.

### 4 Experiments

We implement Moses as our basic SMT system and built it as follows: alignment performed by GIZA++ (Och and Ney, 2003). A phrase-based MT decoder similar to the work of (Koehn et al., 2007) was used with the decoding weights opti-

mized by MERT (Och, 2003). We use a 3-gram language model. Mongolian language resources and Mongolian processing tools are scarce. CWMT’2009 (Zhao et al., 2012) was used for the experiments. It is a small training set when compares to major language training set because as a small language, public Mongolian and Chinese parallel corpus is limit. The lemmatization tool we used is the same as (Yu and Hou, 2011; Hou et al., 2009). Table 3 shows the data set in detail. Mo is the abbreviation of Mongolian and Ch is the abbreviation of Chinese.

	Train	Dev	Test
Bilingual sentence pairs	66808	1000	1000
Scale	18.3MB	214KB	213KB
Total Mo words/stems	869168	11239	11134
Total Ch words	846574	8765	8697
Total Ch characters	1096551	12569	12526

Table 3. Data set

We manually aligned 100 pairs of bilingual sentence to evaluate the alignment performance including precision, recall, F-score and AER (David et al., 2003). As table 4 shows, after using finer-grained stem-based as basic units: precision has been increased from 62.75% to 63.82%; recall has been increased from 75.91% to 83.47% and improved significantly by using Chinese characters; AER has been reduced 2.74% from 39.44 to 38.36. These evaluations prove that our method of using finer-grained for alignment enhances the quality of SMT alignment and reduce the AER. The good performance in alignment partly because of the process of data sparsity we argued in section 2 and partly because of the good realignment we discussed in section 3.

Mongolian	Chinese	Precision	Recall	F-score	AER
word	word	62.75	75.91	69.33	39.44
stem	word	62.94	77.39	70.17	38.83
word	character	63.71	82.25	72.98	38.89
stem	character	63.82	83.47	73.65	38.36

Table 4. Alignment evaluation of Precision, Recall and F-score

To evaluate the translation performance of our method, we do experiments on all kinds of grammatical components including: fully coarser-grained, different grained units for alignment and TRI. We also evaluate the influence on using finer-grained and coarser-grained units on source or target language. In the experiments of translation, we set conventional Mongolian-Chinese SMT system as baseline 1. We also set baseline 2, baseline 3 and baseline 4 which use finer-grained for both alignment and TRI to compare with our systems.

From table 5 we can see that all our three systems outperform the baseline 1. The comparison between our systems and the baseline 1 shows that using finer-grained basic units in alignment outperforms the conventional Mongolian-Chinese SMT. The BLEU of System 3 is higher than system 1 and system 2, which proves that using finer-grained for both source language and target language achieve better performance than using it on one language.

		Alignment	TRI	BLEU
Baseline1	Mo	word	word	21.88
	Ch	word	word	
System 1	Mo	stem	word	22.15
	Ch	word	word	
System 2	Mo	word	word	23.36
	Ch	character	word	
System 3	Mo	stem	word	23.49
	Ch	character	word	

Table 5. Translation evaluation of proposed systems and Baseline 1.

In the comparison of table 6, baseline 2 uses finer-grained basic units for Mongolian alignment and TRI, while system 1 uses finer-grained basic units only for Mongolian alignment but not TRI. System 1 outperformed Baseline 2 indicates that using coarser-grained Chinese units for TRI is more proper and applying our method to source language of Mongolian is successful.

		Alignment	TRI	BLEU
Baseline 2	Mo	stem	stem	21.97
	Ch	word	word	
System 1	Mo	stem	word	22.15
	Ch	word	word	

Table 6. Compare our System 1 with Baseline 2.

In the comparison of table 7, baseline 3 uses finer-grained basic units for Chinese alignment and TRI, while system 2 uses finer-grained basic units only for Chinese alignment but not TRI. System 2 outperformed Baseline 3 indicates that using coarser-grained Chinese units for TRI is more proper and applying our method to target language of Chinese is successful.

		Alignment	TRI	BLEU
Baseline 3	Mo	word	word	23.19
	Ch	character	character	
System 2	Mo	word	word	23.36
	Ch	character	word	

Table 7. Compare our System 2 with Baseline 3.

In the comparison of table 8, baseline 4 uses finer-grained basic units for both Mongolian and Chinese alignment and TRI, while system 3 uses finer-grained basic units only for Mongolian and Chinese alignment but not TRI. System 3 outperformed Baseline 4 indicates that using coarser-grained units in both Chinese and Mongolian for TRI is more proper and our method is successful in the evaluation.

		Alignment	TRI	BLEU
Baseline 4	Mo	stem	stem	22.73
	Ch	character	character	
System 3	Mo	stem	word	23.49
	Ch	character	word	

Table 8. Compare our System 3 with Baseline 4.

These comparisons of table 5 to table 8 proved that:

(1) Using finer-grained for alignment performed better than coarser-grained (table 5) because finer-grained basic units can enhance the alignment quality (table 4).

(2) Using coarser-grained for TRI, which means using finer-grained only for alignment rather than

using them though the whole translation process is better (table 8), because stems and characters are too finer to induct the translation rules.

(3) Using our method of finer-grained for alignment and coarser-grained for TRI improved the conventional SMT system and outperformed other grammatical components (table 5 and table 8);

(4) Using our method only in one side of source language or target language also performed better (table 6 and table 7).

## 5 Conclusion

We presented a method of using finer-grained Mongolian stems and Chinese characters as basic units for alignment, but coarser-grained Mongolian and Chinese words for TRI. Our method outperforms four baselines, mitigates the data sparsity and enhances the alignment quality and translation performance. Through the experiments we find some conclusions as follows: applying finer-grained units can perform a better word alignment result; Using finer-grained basic units for alignment, but coarser-grained for TRI can be a more efficient way than fully finer-grained or fully coarser-grained; using our method for both source language and target language can achieve better performance than using it for either source or target language. We do the same experiments on the Chinese-Mongolian SMT system and get the same conclusion. The experiments indicate that using finer-grained basic units for alignment and coarser-grained basic units for TRI performs better than other granularity combination. We also find that using Chinese characters contribute more than using Mongolian stems in Chinese-Mongolian SMT, which partly because of the errors brought by lemmatization. If we can combine more features (Elming and Habash, 2007) to do the realignment, and have a higher accuracy tool of lemmatization, our method can be better.

## Acknowledgments

We thank PACLIC29 reviewers. This work is supported by the National Natural Science Foundation of China (No. 61362028) and the Postgraduate Scientific Research Innovation Foundation of Inner Mongolia (No. 1402020201).

## Reference

- Peter F. Brown, Stephen A. Della Pietra, Vincent J. Della Peitra, Robert L. Mercer. 1993. The Mathematics of statistical machine translation: parameter estimation. *Computational Linguistics*, 19(2):263-311.
- Pichuan Chang, Michel Galley, and Christopher D. Manning. 2008. Optimizing Chinese word segmentation for machine translation performance. *In proceedings of third workshop on SMT*, pages 224-232.
- Yonggang Deng and Bowen Zhou. 2009. Optimizing word alignment combination for phrase table training. *In Proceedings of the ACL and the 4th IJCNLP of the AFNLP*, pages 229-232
- Vilar, David, Maja Popovic, and Hermann Ney. 2006. AER: Do we need to “improve” our alignments? *In Proceedings of the International Workshop on Spoken Language Translation*, pages 205-212,
- Jakob Elming and Nizar Habash. 2007. Combination of statistical word alignments based on multiple pre-processing schemes. *In Proceedings of the Association for Computational Linguistics*, pages 25-28.
- Wenbin Jiang, Liang Huang, and Qun Liu. 2009. Automatic adaptation of annotation standards: Chinese word segmentation and POS tagging - a case study. *In Proceedings of ACL and the 4th IJCNLP of the AFNLP*, pages 522-530.
- Hongxu Hou, Qun Liu, Nasanurtu. 2009. Mongolian word segmentation based on statistical language model. *Pattern Recognition and Artificial Intelligence*, 22(1): 108-112.
- Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondrej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: Open source toolkit for statistical machine translation. *In ACL*
- Philipp Koehn, Franz Josef Och, and Daniel Marcu. 2003. Statistical phrase-based translation. *In Proceeding of NAACL*.
- Yang Liu, Qun Liu, and Shouxun Lin. 2005. Log-linear models for word alignment. *In ACL '05: Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, pages 459-466.
- Franz Josef Och. 2003. Minimum error rate training in statistical machine translation. *In Proceedings of the Association for Computational Linguistics*, pages 440-447.

- Franz Josef Och and Hermann Ney. 2003. A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29(1): 19-51.
- Ning Xi, Guangchao Tang, Boyuan Li, and Yinggong Zhao. 2011. Word alignment combination over multiple word segmentation. In *Proceedings of the ACL2011 Student Session*, pages 1-5.
- Ning Xi, Guangchao Tang, Xinyu Dai, Shujian Huang, Jiajun Chen. 2012. Enhancing Statistical Machine Translation with Character Alignment. In *Proceedings of the ACL2012*, pages 285-290.
- Xinyan Xiao, Yang Liu, Young-Sook Hwang, Qun Liu, and Shouxun Lin. 2010. Joint tokenization and translation. In *Proceedings of the 23rd International Conference on Computational Linguistics*, pages 1200-1208.
- Ming Yu and Hongxu Hou. 2011. Researching of Mongolian word segmentation system based on dictionary, rules and language model. *Inner Mongolian University*.
- Ruiqiang Zhang, Keiji Yasuda, and Eiichiro Sumita. 2008. Improved statistical machine translation by multiple Chinese word segmentation. In *Proceedings of the Third Workshop on Statistical Machine Translation*, pages 216-223.
- Hongmei Zhao, Yajuan Lv, Guosheng Ben, Yun Huang, Qun Liu. 2012. Summary on CWMT2011 MT Translation Evaluation. *Journal of Chinese Information Processing*, 26(1): 22-30.

# Finding the Origin of a Translated Historical Document

**Zahurul Islam**

MassineScheffer & Company GmbH  
Berlin, Germany  
zahurul.islaam@massine.com

**Natia Dundia**

Department of Modern Languages  
Goethe University Frankfurt, Germany  
dunantuanatia@gmail.com

## Abstract

Gospels are one type of translated historical document. There are many versions of the same Gospel that have been translated from the original, or from another Gospel that has already been translated into a different language. Nowadays, it is difficult to determine the language of the original Gospel from where these Gospels were translated. In this paper we use a supervised machine learning technique to determine the origin of a version of the *Georgian* Gospel.

## 1 Introduction

Translation is a process of rewriting an original text in a different language (Lefevre, 2002). It is one of the oldest text manipulation related processes. Gospels are historical documents that were translated centuries ago. There are many versions of the same Gospel, translated from the original, or from another Gospel that had already been translated into a different language. Nowadays, it is unclear what was the language of the original Gospel from where these were translated. Historians and linguists are uncertain as to the origin of such historical documents. The *Georgian* Gospels are translated from *Armenian* or *Greek* Gospels (Lang, 1957). There are about 300 manuscripts of the four Gospels in *Georgian* that are translated from different languages (Kharanauli, 2000). Linguists are able to narrow down potential origins by looking at different linguistic properties, but skeptical to choose a single origin. We have three such Gospels in *Georgian*, *Armenian* and *Greek*, where linguists believe that *Armenian* or *Greek* are the potential origin. In this paper we use a supervised machine learning technique to find out the correct origin of a version of the *Georgian* Gospel.

One of the challenges of dealing with historical data is the requirement of specific knowledge of

languages that are not spoken at present day. If the language is currently spoken, it is likely that many properties have changed due to language evolution. Due to this issue, the available historical data set is very small in size, which proves a challenge for machine learning algorithms.

From the early stage of translation studies research, translation scholars proposed different kinds of properties of source text and translated text. Recently, scholars in this area identified several properties of the translation process with the aid of corpora (Baker, 1993; Baker, 1996; Olohan, 2001; Laviosa, 2002; Hansen, 2003; Pym, 2005; Toury, 1995). These properties are subsumed under four keywords: *explicitation*, *simplification*, *normalization*, *levelling out* and *interference*.

In this paper, we use texts from modern language to train a Support Vector Machine (SVM) that can be used to identify the original source of the *Georgian* Gospel.

The paper is organized as follows: Section 2 introduced the historical documents that we are dealing with here, Section 3 discusses related work, followed by a discussion of the nature of a translated text in Section 4. The methodology is described in Section 5. The corpus of modern languages is described briefly in Section 6 followed by a discussion of different features we used in this paper in Section 7. The experiment and evaluation in Section 8 and finally, we present conclusions in Section 9.

## 2 The historical documents

Gospels are among the very first documents that were translated into *Georgian* language following the invention of the Georgian alphabet (Lang, 1957). The history begins with the palimpsest manuscripts from the *fifth* or *sixth* centuries and ends with the manuscripts from the *eighteenth* century. There are many open debates on the table about the origin of the Georgian translation of

Language	Sentences	Average Sentence Length	Average Word Length
Georgian	3738	18.96%	4.71%
Armenian	3738	19.15%	4.00%
Greek	3738	20.40%	4.24%

Table 1: Historical corpus statistics

the holy script. According to Blake (1932), many translations were made from the Gospels of *Syrian* and *Armenian*.

However, recent studies show two more sources from where the holy scripture were translated into *Georgian*. The first one is the *Palestinian* and other one is the *Antiochian/Constantinopolitan* (Kharanauli, 2000).

The precise date of these translations are unknown, but the earliest translations of the *Georgian* Bible are presented in the lower script of palimpsests, the so-called *Xanmeti* fragments. *Xanmeti* is a term already used by the famous *Georgian* monk, religious writer and translator *George the Athonite*<sup>1</sup>. He denotes the text where the *x-prefix* is employed to mark the second subject and the third object persons in the *Georgian* verb. This prefix has not occurred in the inscriptions since the seventh Century. Based on philological data, these fragments are dated from *fifth* to *seventh* centuries. Codicological study of the folio size reveals that they are fragments of quite large codices, and it can be assumed that these codices included several books of the Bible.

Currently, there are about 300 manuscripts of the four Gospels in *Georgian* (Kharanauli, 2000). Among these, about 40 codices include text version of *Georgian* Gospels. The Gospel considered for this study is believed to be translated from *Armenian* or *Greek*. These Gospels are digitized and aligned manually. The aligned corpus of the *Georgian* Gospel manuscripts present the texts in their original form side by side, which means that a) nothing is corrected, not even the mistakes presumably made by copyists; and b) abbreviations remain discernible as they are, with the abbreviated letters being indicated in brackets. Table 1 shows the statistics of the Gospels.

### 3 Related work

There is no work found that is exactly relevant to the problem we are dealing here. Lang (1957) studied *Georgian* Gospels and their origins. The first *Georgian* Gospels were translated from an *Ar-*

*menian* version (Lang, 1957). The Gospels that were translated in the late ninth century show signs of revision by reference to the *Greek* Gospels.

Corpus-based translation studies is a recent field of research with a growing interest within the field of computational linguistics. Baroni and Bernardini (2006) started corpus-based translation studies empirically, where they work on a corpus of geo-political journal articles. A SVM was used to distinguish original and translated Italian text using n-gram based features. According to their results, word bigrams play an important role in the classification task.

Van Halteren (2008) uses the *Europarl* corpus for the first time to identify the source language of text for which the source language marker was missing. In their experiments, the support vector regression was the best performing method.

Pastor et al. (2008) and Ilisei et al. (2009; 2010) perform classification of Spanish original and translated text. The focus of their works is to investigate the *simplification* relation that was proposed by (Baker, 1996). In total, 21 quantitative features (e.g. a number of different POS, Average Sentence Length (ASL), the parse-tree depth etc.) were used where, nine (9) of them are able to grasp the simplification translation property.

Koppel and Ordan (2011) have built a classifier that can identify the correct source of the translated text (given different possible source languages). They have built another classifier, which can identify source text and translated text. However, the limitation of this study is that they only used a corpus of English original text and English text translated from various European languages. A list of 300 function words (Pennebaker et al., 2001) was used as feature vector for these classifications.

Popescu (2011) uses *string kernels* (Lodhi et al., 2002) to study translation properties. A classifier was built to classify English original texts and English translated texts from French and German books that were written in the nineteenth century. The *p-spectrum* normalized kernel was used for the experiment. The system performs poorly when the source language of the training corpus is different from the one of the test corpus.

Islam and Hoenen (2013) used a source and translated texts of six European languages in order to classify translated texts according to source languages. As features, they have used the hundred

<sup>1</sup>Wikipedia: [http://en.wikipedia.org/wiki/George\\_the\\_Athonite](http://en.wikipedia.org/wiki/George_the_Athonite)

most frequent words. It is important to consider the properties of language family when dealing with source and translated texts (Islam and Hoenen, 2013).

Features used by Koppel and Ordan (2011) and Islam and Hoenen (2013) are language dependent. As we use texts from twenty-one European languages to build the training model, we only use features that are language and linguistic tools independent. It is also important to consider different properties of translated and source texts proposed by translation scholars.

## 4 Translation properties

Recently, translation scholars proposed different translation properties using monolingual or comparable corpus. These properties are described in the following subsections.

### 4.1 Explicitation

Translators are biased to make translations more *explicit* in order to resolve ambiguities that might be inherited in the translated text. Vinay and Darbelnet (1958) used the term *explicitation* as “a process of introducing information into the target language which is present only implicitly in the source language, but which can be derived from the context or situation” (Vinay and Darbelnet, 1995; Pym, 2005). However, Blum-Kulka (1986) first claimed *explicitation* as a translation universal where she studied translated *French* texts from *English* by professional and non-professional translators. Seguinot (1988) provides an empirical study using two translated texts from *French* to *English*. There is a greater level of *explicitness* in the translated texts as linking words and conversion of subordinate clauses into coordinate clauses.

### 4.2 Simplification

The *simplification* translation property shows the tendency of a translator to simplify a text in order to improve the readability of a translated text. Blum-Kulka and Levenston (1978) mention the term *simplification* as part of the lexical simplification using a small data set of *English* and *Hebrew* translations. According to them, translators use techniques such as *avoidance* and *approximation* in the translation process to make a translated text simpler for the target readers. Later, Baker (1996) also observed this tendency in the translated texts.

To make a translated text simpler, the translator often breaks up complex sentences into two or more sentences. This tendency can be found in the ASL. That is, the ASL in a translated text will be shorter than a source text.

### 4.3 Normalization

The *normalization* property shows a translator’s effort to meet the normative criteria of the target language. It is a translator’s tendency to conform to patterns and practices that are typical of the target language, even to exaggerate their use. This property can be observed in a translated text that contains very little trace of the source language. However, the opposite scenario can be seen as well, where the translation is influenced by the source language. In that case *normalization* will be weakened. The influence of *English* can be visible in many software manuals that are translated from *English*. Hansen (2003) stated that this contrary tendency also can be seen in *interpreting*, where the interpreter tries to finish an unfinished sentence and to render an ungrammatical structure into something grammatical.

### 4.4 Levelling out

Baker (1996) refers to *levelling out* as “the tendency of translated text to gravitate towards the center of a continuum”. That is also known as *convergence* (Laviosa, 2002), where she stated that a “relatively higher level of homogeneity of translated texts with regard to their own scores on given measures of universal features” such as lexical density or sentence length, in contrast to source texts. If we have a sub-corpus of translated texts from different languages to the same language, and have source texts in the same language, then translated texts from different languages will be similar in terms of *lexical density*, TTR, and ASL; but will be different than the source texts. More specifically, translated texts from different languages will be alike but will be different than the source texts.

### 4.5 Interference

Toury (1995) has a different theory that is different from the translation properties described above. He stated that “in translation, phenomena pertaining to the make-up of the source text tend to be transferred to the target text,” That is, some *interference* effects will be observable in translated

texts that are carried from source texts. These effects will be in the form of *negative transfer* or in the form of *positive transfer*. As an example, specific properties of the English language are visible in user manuals that have been translated to other languages from English (for instance, word order) (Lzwaini, 2003). We can summarize this translation properties in a way that a translated text from different source languages will be sufficiently different from each other.

## 5 Methodology

The above section describes the properties of translation. Based on these properties, a translated text is different than the corresponding source text. Properties proposed by translation scholars, focus on texts and the translation process. Our assumption is that even though historical texts were translated many hundreds of years ago, there are some properties that are common to modern texts and the recent translation process.

We model the task as a classification task where we use a SVM implementation to find the correct origin of the *Georgian* Gospel. Linguists believe that the *Georgian* Gospel is a translated document. They narrowed down potential origins by looking at different linguistic properties compared to the *Greek* and *Armenian* Gospel. Before finding the source of the *Georgian* Gospel, it is necessary to check that the Gospel itself is a translated document. If the gospel is classified as a translated document then we can move further to find the source. The gospel that has properties of an original document will be the closest candidate for the origin *Georgian* gospel.

In order to build a training model, we use modern texts from different European languages. We have compiled a suitable corpus for this task from the *Europarl* corpus (Koehn, 2005). This task requires features that are language independent and do not require any linguistic pre-processing. So, we have explored different features that are quantitative indicators of translation properties mentioned above. Finally, we have collected a list of useful features that are listed in Section 7. We use standard classification *accuracy* and *F-Score* in order to measure usefulness of a feature. At the beginning the feature list contains only ASL. We have added a new feature in the list if and only if the classification *accuracy* and *F-Score* improve by adding the feature with existing feature set. The

feature collection process will be continued until the classifier achieves a reasonable *accuracy F-Score*. Figure 1 shows the approach we follow in this paper. Finally, the whole corpus of modern texts will be considered for building the final training model.

The final training model and the collected feature set will be used in order to find the origin of the *Georgian* Gospel. We prepare the Gospels data into two sets similarly as the training data. The first set of data will contain texts from *Armenian* and *Georgian* Gospels and the other one will contain texts from *Greek* and *Georgian* Gospels.

## 6 Corpus of modern texts

The area of translation studies lack corpora by which scholars can validate their theoretical claims, for example, regarding the scope of characteristics of the translation properties. This scope is obviously affected by the membership of the source and target languages to language families. Though the exploration of universally valid characteristics of translations is an important topic, there are not many resources for testing corresponding hypotheses.

There are many parallel and multilingual corpora available nowadays. Most of them are not useful for translation studies immediately as they require customization. Islam and Mehler (2012) provide a customized resource in which the languages of all source texts and their translations are annotated sufficiently. The resource they provide is a customized version of the well-known *Europarl* corpus (Koehn, 2005). A central feature of this corpus is that it provides information on sentence-related alignments that can be explored for finding characteristics of the translation relation.

The language annotation in the *Europarl* corpus is not reliable because of erroneous annotations introduced by translators. There are many cases where one speaker has multiple speeches in different languages that cause problems for identifying the speaker's native language.

In order to resolve this issue we have collected the name of the member of the *European parliament* and their native language manually. We collected names from the current members list page <sup>2</sup> of the *European parliament*. Names of former members are collected from the correspond-

<sup>2</sup><http://www.europarl.europa.eu/meps/en/full-list.html>

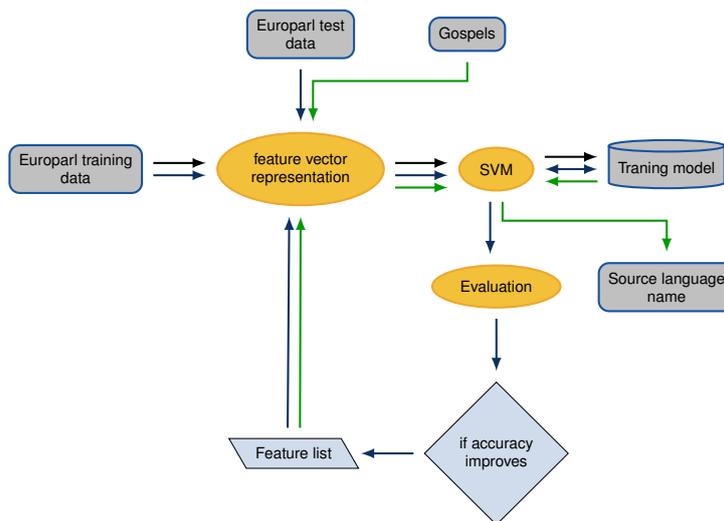


Figure 1: Machine learning approach to find the source of the Georgian Gospel

ing *Wikipedia* pages. The official language of the country of each member is assigned as the native language of a speaker. Members from *Belgium* and *Luxembourg* are not considered as we are not sure about the language spoken by members from these countries in the *European parliament*. Each member from *Finland* is assigned to the *Finnish* language. Finally, the list contains 2,125 member names and their native language. This list is used to extract source and translated texts from the *Europarl* corpus. The corpus contains 2,646,765 parallel sentences from 412 language pairs of 21 European languages. We believe that such a corpus is an ideal resource for the problem we are addressing in this paper.

## 7 Features

As the training corpus contains texts from twenty-one European languages, we only experiment with *lexical* and *information-theoretic* features. Pastor et al. (2008) used various *lexical*, *syntactic* and *discourse* related features. Also, Ilisei et al. (2009; 2010) used similar type of features. The following sub sections describe features that are finally selected for the feature list.

### 7.1 Lexical features

Different lexical features are being used from the beginning of corpus based translation studies. These features are popular for other NLP applications such as *text readability classification*. The reason behind the popularity is that these are language independent and do not require any linguistic pre-processing.

The ASL is a quantitative measure of syntactic complexity. Generally, the syntax of a longer sentence is more complex than that of a shorter sentence. A translator tries to make a translation *explicit* and also *simple*. Translated texts might become longer due to the *explicitation*. However, opposite can happen when a translator tries to make a translation simpler. Table 2 shows behavior of some features in source and translated texts of four European languages. Translations of German, French and Dutch are more explicit than Spanish. The Average Word Length (AWL) is another useful lexical feature. Most of the cases, the AWL in translated texts is longer than source texts. It would be interesting to see the behavior of AWL in source and translated texts of an agglutinative language.

The *Average number of complex words* feature is related to the AWL. A translated text will be difficult for readers if it contains more complex words. The average length of English written words is 5.5 (Nádas, 1984) letters. We define a *complex word* as any word that contains 10 or more letters.

The Type Token Ratio (TTR), which indicates the lexical density of text, has been considered as useful features by Pastor et al. (2008) and Also, Ilisei et al. (2009; 2010). Low lexical densities involve a great deal of repetition with the same words occurring again and again. Conversely, high lexical density shows the diverseness of a text. A diverse text is supposed to be difficult for readers, generally children (Temnikova, 2012). There are many different version of TTR formulas avail-

	ASL		AWL		Entropy	
	Source	Translation	Source	Translation	Source	Translation
German	26.07	29.34	5.52	5.64	9.95	9.58
French	33.86	34.46	4.65	4.68	9.43	9.12
Spanish	35.99	32.56	4.66	4.74	9.08	9.02
Dutch	25.43	31.13	4.88	5.08	9.30	8.99

Table 2: Observation of different features

able. Carrol (1964) proposed a variation of TTR in order to reduce the sample size effect. Another version of TTR is called Bilogarithmic TTR (Herdan, 1964). Kohler and Galle (1993) also defined a version TTR (see: 1) that consider position of the text. In the Equation 1  $x$  refers to position in the text,  $t_x$  = number of types up to position  $x$ ,  $T$  = number of types in the text and  $N$  refers to the number of tokens in the whole text. We also used another version of TTR that focuses on document level TTR  $\frac{T}{N}$  as well as sentence level TTR  $\frac{t}{n}$  (Islam and Mehler, 2013; Islam, 2014; Islam et al., 2014). Lower TTR in sentence level also shows the repetition of the text.

- Köhler–Gale method

$$TTR_x = \frac{t_x + T - \frac{x^T}{N}}{N} \quad (1)$$

- Root TTR

$$\frac{T}{\sqrt{N}} \quad (2)$$

- Corrected TTR

$$\frac{T}{\sqrt{2N}} \quad (3)$$

- Bilogarithmic TTR

$$\frac{\log T}{\log N} \quad (4)$$

- TTR deviation

$$\sum_{i=0}^n \left( \frac{T}{N} - \frac{t_i}{n_i} \right) \quad (5)$$

## 7.2 Information-theoretic features

Information theory measures the statistical significance of how documents vary with different types of probability distributions. That is, it determines how much information can be encoded from a document using a certain type of probability distribution. The use of information as a statistical measure of significance is an extension of this process. Information theory allows us to use conditional probabilities. It should be noted that these features are being used for the first time on this kind of problem.

### 7.2.1 Entropy based features

The most efficient way to send information through a noisy channel is at a constant rate (Genzel and Charniak, 2002; Genzel and Charniak, 2003; ?). This rule must be retained in any kind of communication to make it efficient. Any text as a medium of communication should satisfy this principle. Genzel and Charniak (2002; 2003) show that the entropy rate is constant in texts. That is, for example, each sentence of a text conveys roughly the same amount of information. In order to utilize this information-theoretic notion, we start from random variables and consider their entropy as indicators of readability.

Shannon (1948) introduced entropy as a measure of information. Entropy, the amount of information in a random variable, can be thought of as the average length of the message needed to have an outcome on that variable. The entropy of a random variable  $X$  is defined as

$$H(X) = - \sum_{i=1}^n p(x_i) \log p(x_i) \quad (6)$$

The more the outcome of  $X$  converges towards a uniform distribution, the higher  $H(X)$ . Our hypothesis is that the higher the entropy, the less readable the text along the feature represented by  $X$ . Table 2 shows that translated texts have lower entropy than source texts. This is because translators try to improve the readability of translated texts. In our experiment, we consider the following random variables: *word probability*, *character probability*, *word length probability* and *word frequency probability* (or frequency spectrum, respectively). Note that there is a correlation between the probability distribution of words and the corresponding distribution of word frequencies. As we use SVM for classification, these correlations are taken into consideration.

### 7.3 Information Transmission-based Features

There is a relation among text difficulty, sentence length, and word length. The usefulness of similar

lexical features such as *sentence length* or *number of difficult words in a sentence* is shown in section 7.1. Generally, a longer sentence contains more entities that influence the difficulty level. Similar things happen with longer words. But, a sentence becomes more difficult if it is longer and contains more long words. These kinds of properties can be defined by *joint* and *conditional* probabilities.

In the field of information theory, joint probability measures the likelihood of two events occurring together. That is, two random variables  $X$  and  $Y$  will be defined in the probability space. The conditional probability gives the probability that the event will occur given the knowledge that another event has already occurred. By considering the joint probability and two random variables  $X$  and  $Y$ , Shannon's joint entropy can be defined as:

$$H(X, Y) = - \sum_{\langle x, y \rangle \in X \times Y} p(x_i, y_i) \log p(x_i, y_i) \quad (7)$$

Two conditional entropies can be defined as:

$$H(X|Y) = - \sum_{y \in Y} P(y_i) \sum_{x \in X} p(x_i|y_i) \log p(x_i|y_i) \quad (8)$$

$$H(Y|X) = - \sum_{x \in X} P(x_i) \sum_{y \in Y} p(y_i|x_i) \log p(y_i|x_i) \quad (9)$$

From the equation 6, 7, 8 and 9, it can be shown that:

$$T_s(X, Y) = H(X) + H(Y) - H(X, Y) \quad (10)$$

The function is called *Information transmission*, and it measures the strength of the relationship between elements of random variables  $X$  and  $Y$ . Details about this notion can be found in (Klir, 2005). The *sentence length and word length probability* shows the relation between sentence length and word length and *sentence length and difficult word probability* shows the relation between sentence length and the number of difficult words.

## 8 Experiment

The experiments and evaluations are explained in the following subsections.

### 8.1 Experiment with modern corpus

The training corpus contains 2,646,765 parallel sentences from 412 language pairs of 21 European languages. We have divided the corpus into 26,467 chunks. More specifically, 26,467 chunks were *source* texts and the same number of chunks were *translations*. It should be noted that a hundred sets of data were randomly generated where 80% of the corpus is used for training and the remaining 20% is used for evaluation. Later, when we get reasonable classification *accuracy* and *F-Score*, the whole corpus will be used to build the final training model. The weighted average of *Accuracy* and *F-score* is computed by considering all sets of data. Note that we have used the SMO (Platt, 1998; Keerthi et al., 2001) classifier model in WEKA (Hall et al., 2009) together with the Pearson VII function-based universal kernel PUK (Üstün et al., 2006).

As we showed in Figure 1, our goal was to build a model using texts from modern European languages and later use that model to identify the source of the *Georgian Gospel*. The challenge was to find features that are language independent and improve the classification accuracy. A feature will be in the feature list if and only if the classification accuracy improves by adding the feature. Many different features were considered, but only useful features are listed in Table 3 and described in Section 7. Additionally, either measure *Accuracy* and *F-score* has to be above average. Individually all features perform reasonably well. However, *information-theoretic* features perform better than lexical features. Table 3 shows evaluation of selected features. Surprisingly *word frequency entropy* is the best performing individual feature. Altogether these features achieve 86.62% of *F-Score*.

### 8.2 Experiment with target corpus

In order to experiment with the target corpus, we prepare them similarly to the training chunks. Each Gospel was divided into 37 chunks. Each chunk contains 100 verses. Then, these data are divided into two sets. The first set contains chunks from *Armenian* and *Georgian*. The other contains chunks from *Greek* and *Georgian*.

As we stated earlier, the first task is to identify chunks of the *Georgian Gospel* are translations. Table 4 shows the confusion matrix of the first set. In this matrix 36 out of 37 chunks of

Feature	Accuracy	F-Score
ASL	54.01%	53.29%
TTR per document	59.83%	59.18%
TTR per sentence	58.93%	57.42%
Average complex word per document	52.61%	45.74%
Average complex word per sentence	52.52%	45.83%
AWL	56.15%	49.43%
Köhler–Gale TTR	59.58%	58.89%
Root TTR	62.67%	62.67%
Corrected TTR	62.61%	62.61%
Bi-logarithmic TTR	62.23%	62.08%
TTR deviation	60.54%	60.00%
Word entropy	62.02%	61.92%
Word frequency entropy	63.36%	63.39%
Word length entropy	53.81%	50.94%
Character entropy	57.78%	56.58%
Character frequency entropy	57.93%	57.28%
Information transmission of sentence length and word length probability	52.93%	50.26%
Information transmission of sentence length and complex word probability	54.41%	53.86%
All features	86.63%	86.62%

Table 3: Evaluation of lexical features in source and translation identification

	Source	Translation
Armenian	0	37
Georgian	1	36

Table 4: Confusion matrix of *Armenian–Georgian* Gospels

	Source	Translation
Greek	20	17
Georgian	1	36

Table 5: Confusion matrix of *Greek–Georgian* Gospels

the *Georgian* Gospel identified as translated text. So, experimental results show that the *Georgian* Gospel is a translated document. Table 5 shows the same result. All of the *Armenian* chunks are identified as translated documents. However, 20 out of 37 chunks of the *Greek* Gospel are identified as source. Therefore, these two confusion matrices show that *Greek* is most likely the source of the *Georgian* Gospel. It becomes clearer when we have a look on Table 6. Here *Armenian* and *Greek* chunks are labeled as *source* and *Georgian* chunks are labeled as *translation*. The *accuracy* and *F-Score* of the *Armenian–Georgian* pair is below the baseline 50%. But the *accuracy* and *F-Score* of the *Greek–Georgian* pair is above 75%. So, our experimental results suggest that the *Greek* Gospel is the source of the version of *Georgian* Gospel.

Source-translation	Accuracy	F-Score
Armenian–Georgian	48.64%	32.73%
Greek–Georgian	75.67%	74.48%

Table 6: Classification results of Gospels

## 9 Conclusion

It is important to identify a document as original or translated from another language. Such a tool is very useful for many NLP applications. Different linguistic features are being explored in recent days for many different NLP applications. However, only simple *lexical* and classical *information-theoretic* features are adequate to build a classifier which is able to identify an original or a translated document. It will be challenging to explore linguistic features for such applications that deal with multilingual data.

There are many versions of the *Georgian* Gospels that are translated from different languages. Linguists are able to narrow down potential origins by looking at different linguistic properties, but skeptical to decide the single origin. We have three such Gospels in *Georgian*, *Armenian* and *Greek*, where linguists believe that *Armenian* or *Greek* are the potential origin. For this paper, we have built a source and translation classifier using modern texts. The classifier is able to identify translated documents that have been translated hundreds of years ago. Based on our experimental evaluation, the *Greek* Gospel is the source of the version of the *Georgian* Gospel.

## 10 Acknowledgments

We would like to thank Prof. Dr. Alexander Mehler, Prof. Dr. David Scheffer and Armin Hoenen. We also like to thank the anonymous reviewers for their helpful comments. This work was performed when both authors were part of the LOEWE Digital-Humanities project at Goethe-University Frankfurt. Travel was funded by MassineScheffer & Company GmbH.

## References

- Mona Baker. 1993. Corpus linguistics and translation studies - implications and applications. In Mona Baker, Gill Francis, and Elena Tognini-Bonelli, editors, *Text and Technology. In Honour of John Sinclair*, pages 233–354. John Benjamins.
- Mona Baker. 1996. Corpus-based translation studies: The challenges that lie ahead. In *LSP and Translation: Studies in Language Engineering in Honour of Juan C. Sager*, pages 175–186. Amsterdam & Philadelphia: John Benjamins.
- Marco Baroni and Silvia Bernardini. 2006. A new approach to the study of translationese: Machinelearning the difference between original and translated text. *Literary and Linguistic Computing*, 21(3):259–274.
- Robert Pierpont Blake. 1932. *Khanmeti palimpsest fragments of the Old Georgian version of Jeremiah*. Cambridge Univ Press.
- Shoshana Blum and Eddie A Levenston. 1978. Universals of lexical simplification. *Language learning*, 28(2):399–415.
- Shoshana Blum-Kulka. 1986. Shifts of cohesion and coherence in translation. *Interlingual and intercultural communication: Discourse and cognition in translation and second language acquisition studies*, pages 17–35.
- John Bissell Carroll. 1964. *Language and thought*. Prentice-Hall Englewood Cliffs, NJ.
- Dimitry Genzel and Eugene Charniak. 2002. Entropy rate constancy in text. In *Proceedings of the 40st Meeting of the Association for Computational Linguistics (ACL 2002)*.
- Dimitry Genzel and Eugene Charniak. 2003. Variation of entropy and parse trees of sentences as a function of the sentence number. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*.
- Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, and Ian H. Witten. 2009. The WEKA data mining software: an update. *ACM SIGKDD Explorations*, 11(1):10–18.
- Silvia Hansen. 2003. *The Nature of Translated Text: An Interdisciplinary Methodology for the Investigation of the Specific Properties of Translations*. Ph.D. thesis, University of Saarland.
- Gustav Herdan. 1964. *Quantitative linguistics*. Butterworths.
- Iustina Ilisei, Diana Inkpen, Gloria Corpas Pastor, and Ruslan Mitkov. 2009. Towards simplification: A supervised learning approach. In *Proceedings of Machine Translation 25 Years On, London, United Kingdom, November 21-22*.
- Iustina Ilisei, Diana Inkpen, Gloria Corpas Pastor, and Ruslan Mitkov. 2010. *Identification of translationese: A machine learning approach*, pages 503–511. Springer.
- Zahurul Islam and Armin Hoenen. 2013. Source and translation classification using most frequent words. In *6th International Joint Conference on Natural Language Processing (IJCNLP)*.
- Zahurul Islam and Alexander Mehler. 2012. Customization of the europarl corpus for translation studies. In *Proceedings of the 8th International Conference on Language Resources and Evaluation (LREC)*.
- Zahurul Islam and Alexander Mehler. 2013. Automatic readability classification of crowd-sourced data based on linguistic and information-theoretic features. In *14th International Conference on Intelligent Text Processing and Computational Linguistics*.
- Md. Zahurul Islam, Md. Rashedur Rahman, and Alexander Mehler. 2014. Readability classification of bangla texts. In *15th International Conference on Intelligent Text Processing and Computational Linguistics (icLing), Kathmandu, Nepal*.
- Zahurul Islam. 2014. Multilingual text classification using information theoretic features. PhD Thesis, Goethe University Frankfurt.
- S.S. Keerthi, S. K. Shevade, C. Bhattacharyya, and K. R. K. Murthy. 2001. Improvements to Platt’s SMO algorithm for SVM classifier design. *Neural Computation*, 13(3):637–649.
- Anna Kharanauli, 2000. *Einführung in die georgische Psalterübersetzung*, pages 248–308. Vandenhoeck & Ruprecht.
- George Jiri Klir. 2005. *Uncertainty and Information*. Wiley-Interscience.
- Philipp Koehn. 2005. Europarl: A parallel corpus for statistical machine translation. In *MT Summit*.
- Reinhard Köhler and Matthias Galle. 1993. Dynamic aspects of text characteristics. *Quantitative text analysis*, pages 46–53.
- Moshe Koppel and Noam Ordan. 2011. Translationese and its dialects. In *49th Annual Meeting of the Association for Computational Linguistics (ACL)*.
- David Marshall Lang. 1957. Recent work on the georgian new testament. *Bulletin of the School of Oriental and African Studies*, 19(01):82–93.
- Sara Laviosa. 2002. *Corpus-based translation studies. Theory, findings, applications*. Amsterdam/New York: Rodopi.
- André Lefevre. 2002. *Translation/history/culture: A sourcebook*. Routledge.

- Huma Lodhi, Craig Saunders, John Shawe-Taylor, Nello Cristianini, and Chris Watkins. 2002. Text classification using string kernels. *The Journal of Machine Learning Research*, 2:419–444.
- Sattar Lzwaini. 2003. Building specialised corpora for translation studies. In *Workshop on Multilingual Corpora: Linguistic Requirements and Technical Perspectives, Corpus Linguistics*.
- A. Nádas. 1984. Estimation of probabilities in the language model of the ibm speech recognition system. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 32(4):859–861.
- Maeve Olohan. 2001. Spelling out the optionals in translation: a corpus study. In *Corpus Linguistics 2001 conference. UCREL Technical Paper number 13. Special issue*.
- Gloria Corpas Pastor, Ruslan Mitkov, Naveed Afzal, and Viktor Pekar. 2008. Translation universals: do they exist? a corpus-based NLP study of convergence and simplification. In *Proceedings of the Eighth Conference of the Association for Machine Translation in the Americas (AMTA-08)*.
- Jams W. Pennebaker, Martha E. Francis, and Roger J. Booth. 2001. *Linguistic Inquiry and Word Count (LIWC): LIWC2001 Manual*. Erlbaum Publishers.
- John C. Platt. 1998. *Fast training of support vector machines using sequential minimal optimization*. MIT Press.
- Marius Popescu. 2011. Studying translationese at the character level. In *Recent Advances in Natural Language Processing*.
- Anthony Pym. 2005. Explaining explicitation. In *New Trends in Translation Studies. In Honour of Kinga Klauy*, pages 29–34. Akademia Kiad.
- Candace Séguinot. 1988. Pragmatics and the explicitation hypothesis. *TTR: traduction, terminologie, rédaction*, 1(2):106–113.
- Claude Elwood Shannon. 1948. A mathematical theory of communication. *The Bell System Technical Journal*, 27(1):379–423.
- Irina Temnikova. 2012. *Text Complexity and Text Simplification in the Crisis Management Domain*. Ph.D. thesis, University of Wolverhampton.
- Gideon Toury. 1995. *Descriptive Translation Studies and Beyond*. John Benjamins, Amsterdam/Philadelphia.
- B. Üstün, W.J. Melssen, and L.M.C. Buydens. 2006. Facilitating the application of support vector regression by using a universal Pearson VII function based kernel. *Chemometrics and Intelligent Laboratory Systems*, 81(1):29–40.
- Hans Van Halteren. 2008. Source language markers in europarl translations. In *International Conference in Computational Linguistics (COLING)*, pages 937–944.
- Jean-Paul Vinay and Jean Darbelnet. 1958. Stylistique comparée de l'anglais et du français.
- Jean-Paul Vinay and Jean Louis Darbelnet. 1995. *Comparative stylistics of French and English: a methodology for translation*, volume 11. John Benjamins.

# Improving the Performance of an Example-Based Machine Translation System Using a Domain-specific Bilingual Lexicon

Nasredine Semmar, Othman Zennaki, Meriama Laib  
CEA, LIST, Vision and Content Engineering Laboratory  
F-91191, Gif-sur-Yvette, France

{nasredine.semmar, othman.zennaki, meriama.laib}@cea.fr

## Abstract

In this paper, we study the impact of using a domain-specific bilingual lexicon on the performance of an Example-Based Machine Translation system. We conducted experiments for the English-French language pair on in-domain texts from Europarl (European Parliament Proceedings) and out-of-domain texts from Emea (European Medicines Agency Documents), and we compared the results of the Example-Based Machine Translation system against those of the Statistical Machine Translation system Moses. The obtained results revealed that adding a domain-specific bilingual lexicon (extracted from a parallel domain-specific corpus) to the general-purpose bilingual lexicon of the Example-Based Machine Translation system improves translation quality for both in-domain as well as out-of-domain texts, and the Example-Based Machine Translation system outperforms Moses when texts to translate are related to the specific domain.

## 1 Introduction

There are mainly two approaches for Machine Translation (MT): rule-based and corpus-based (Trujillo, 1999; Hutchins, 2003). Rule-Based MT (RBMT) approaches require manually made bilingual lexicons and linguistic rules, which can be costly, and not generalized to other languages. Corpus-based machine translation approaches are effective only when large amounts of parallel corpora are available. However, parallel corpora are only available for a limited number of language

pairs and domains. In several fields, available corpora are not sufficient to make Statistical Machine Translation (SMT) approaches operational. Most previous works addressing domain adaptation in machine translation have proven that a SMT system, trained on general texts, has poor performance on specific domains. In this paper, we study the impact of using a domain-specific bilingual lexicon on the performance of an Example-Based Machine Translation (EBMT) system, and we compare the results of the EBMT system against those of the SMT system Moses on in-domain and out-of-domain texts.

The rest of the paper is organized as follows. In Section 2, we present previous research in the field of domain adaptation in SMT. Section 3 describes the translation process and the main components of the EBMT system. Section 4 presents the experimental setup and inspects the results of the EBMT system in qualitative and quantitative evaluations. Section 5 concludes our study and presents our future research directions.

## 2 Related Work

Domain adaptation consists in adapting MT systems designed for one domain to work in another. Several ideas have been explored and implemented in domain adaptation of SMT (Bungum and Gambäck, 2011). Langlais (2002) integrated domain-specific lexicons in the translation model of a SMT engine which yields a significant reduction in word error rate. Lewis et al. (2010) developed domain specific SMT by pooling all training data into one large data pool, including as much in-domain parallel data as possible. They trained highly specific language models on in-domain monolingual data in order to reduce the dampening effect of heterogeneous data on quality within the domain. Hildebrand et al.

(2005) used an approach which consisted essentially in performing test-set relativization (choosing training samples that look most like the test data) to improve the translation quality when changing the domain. Civera and Juan (2007), and Bertoldi and Federico (2009) used monolingual corpora and Snover et al. (2008) used comparable corpora to adapt MT systems designed for Parliament domain to work in News domain. The obtained results showed significant gains in performance. Banerjee et al. (2010) combined two separate domain models. Each model is trained from small amounts of domain-specific data. This data is gathered from a single corporate website. The authors used document filtering and classification techniques to realize the automatic domain detection. Daumé III and Jagarlamudi (2011) used dictionary mining techniques to find translations for unseen words from comparable corpora and they integrated these translations into a statistical phrase-based translation system. They reported improvements in translation quality (between 0.5 and 1.5 BLEU points) on four domains and two language pairs. Pecina et al. (2011) exploited domain-specific data acquired by domain-focused web-crawling to adapt general-domain SMT systems to new domains. They observed that even small amounts of in-domain parallel data are more important for translation quality than large amounts of in-domain monolingual data. Wang et al. (2012) used a single translation model and generalized a single-domain decoder to deal with different domains. They used this method to adapt large-scale generic SMT systems for 20 language pairs in order to translate patents. The authors reported a gain of 0.35 BLEU points for patent translation and a loss of only 0.18 BLEU points for generic translation.

### 3 The Translation Process of the Example-Based Machine Translation System

The translation process of the EBMT system consists of several steps: retrieving translation candidates from a monolingual corpus using a cross-language search engine, producing translation hypotheses using a transducer, using word lattices to represent the combination of translation candidates and translation hypotheses, and choosing the *n*-best translations according to a statistical language model learned from a target

language corpus (Semmar and Bouamor 2011; Semmar et al., 2011; Semmar et al., 2015). This process uses a cross-language search engine, a bilingual reformulator (transducer) and a generator of translations. In order to illustrate the functioning of the EBMT system, we indexed a small textual database composed of 1127 French sentences extracted from the ARCADE II corpus (Veronis et al., 2008) and we considered the input source sentence "Social security funds in Greece encourage investment in innovation." as the sentence to translate.

#### 3.1 The Cross-language Search Engine

The role of the cross-language search engine is to extract for each sentence to translate (user's query) sentences or sub-sentences from an indexed monolingual corpus in the target language (Davis and Ogden, 1997; Grefenstette, 1998; Baeza-Yates and Ribeiro-Neto, 1999). These sentences or sub-sentences correspond to a total or a partial translation of the sentence to translate. The cross-language search engine used in the EBMT system is based on a deep linguistic analysis of the query and the monolingual corpus to be indexed and uses a weighted vector space model (Salton and McGill, 1986; Besançon et al., 2003; Semmar et al., 2006). This cross-language search engine is composed of the following modules:

- A linguistic analyzer based on the open source multilingual platform LIMA<sup>1</sup> (Besançon et al., 2010) which includes a morphological analyzer, a Part-Of-Speech tagger and a syntactic analyzer. This analyzer processes both sentences to be indexed in the target language and the sentence to translate in order to produce a set of normalized lemmas with their linguistic information (Part-Of-Speech, gender, number, etc.). The syntactic analyzer implements a dependency grammar to produce syntactic dependencies relations (used to compute compound words) and works by identifying verbal and nominal chains. These syntactic dependencies are detected using finite-state automata defined by rules expressing possible successions of grammatical categories.

<sup>1</sup> <https://github.com/aymara/lima>.

- A statistical analyzer that attributes to each word or a compound word of the sentences to be indexed a weight. For this purpose, we use the TF-IDF weighting. The weight  $w_{ij}$  of term  $j$  in document  $i$  is defined with the formula  $w_{ij}=tf_{ij}logN/n_j$ , where  $tf_{ij}$  is the frequency of term  $j$  in document  $i$ ,  $N$  is the total number of documents in the collection, and  $n_j$  is the number of documents where term  $j$  appears.
- An indexer to build the textual database which contains the sentences of the target language.
- A query reformulator to expand queries during the interrogation of the textual database. The query terms are translated using a bilingual lexicon. Each term of the query is reformulated into its translations in target language using an English-French lexicon composed of 243539 entries<sup>2</sup>.
- A comparator which measures the similarity between the sentence to translate (query) and the indexed sentences in order to retrieve the closest sentences to the reformulated query. The *Cosine* similarity is used to measure the distance between the sentence to translate and each sentence of the textual database. The retrieved sentences are classified by the comparator which groups in the same cluster the sentences that share the same words.

For example, from the sentence “Social security funds in Greece encourage investment in innovation.”, two nominal chains are recognized: “Social security funds in Greece” and “investment in innovation”. From the first nominal chain, the syntactic analyzer recognizes three compound words: Social security funds in Greece (Greece\_fund\_security\_social), Social security funds (fund\_security\_social), and Social security (security\_social). Table 1 illustrates the two first translation candidates provided by the cross-language search engine for the sentence to translate “Social security funds in Greece encourage investment in innovation.”. These sentences share with the query the terms “fund\_security\_social, Greece, investment” for the first class and the term

“fund\_security\_social” for the second class. In addition, the cross-language search engine provides the linguistic information (lemma, Part-Of-Speech, gender, number and syntactic dependency relations) of all words included in the translation candidates (Table 2). The translation candidates are represented as graphs of words and encoded with Finite-State Machines (FSMs). Each transition of the automaton corresponds to the lemma and its linguistic information which is provided by the linguistic analyzer of the cross-language search engine (Figure 1).

Class n°.	Class query terms	Translation candidates
1	fund_security_social, Greece, investment	Les caisses de sécurité sociale de Grèce revendiquent l'indépendance en matière d'investissements.
2	fund_security_social	Objet: Caisses de sécurité sociale grecques.

Table 1. The two first translation candidates returned by the cross-language search engine for the sentence of to translate "Social security funds in Greece encourage investment in innovation."

Les [le, *Plural determiner*] caisses [caisse, *Plural common noun*] de [de, *Singular preposition*] sécurité [sécurité, *Singular common noun*] sociale [social, *Singular adjective*] de [de, *Singular preposition*] Grèce [Grèce, *Singular proper noun*] revendiquent [revendiquer, *Third person plural verb*] l'[le, *Singular determiner*] indépendance [indépendance, *Singular common noun*] en [en, *Singular preposition*] matière [matière, *Singular common noun*] d'[de, *Singular preposition*] investissements [investissement, *Plural common noun*]. [., *Punctuation*]

Table 2: Linguistic information (lemma, grammatical category) of the words of the first translation candidate. This sentence is composed of two nominal chains linked by the word “revendiquent”.

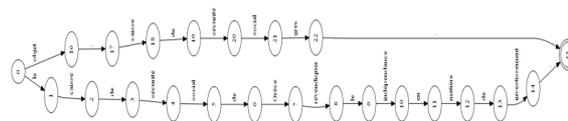


Figure 1: FSMs representing the retrieved sentences returned by the cross-language search engine.

<sup>2</sup> [http://catalog.elra.info/product\\_info.php?products\\_id=666](http://catalog.elra.info/product_info.php?products_id=666).

### 3.2 The Bilingual Reformulator

The role of the bilingual reformulator is to produce a set of translation hypotheses from the sentence to translate. It consists, on the one hand, in transforming into the target language the syntactic structure of the sentence to translate, and, on the other hand, in translating its words. The reformulator uses a set of linguistic rules to transform syntactic structures from the source language to the target language (Syntactic transfer) and the bilingual lexicon of the cross-language search engine to translate words of the sentence to translate (Lexical transfer). The rules of the syntactic transfer are built manually and are based on morpho-syntactic patterns (Table 3). Expressions (phrases) corresponding to each pattern are identified by the syntactic analyzer during the step of recognition of verbal and nominal chains. These expressions can be seen as sentences accepted by a FSM transducer whose outputs are instances of these sentences in the target language (Figure 2).

Rule n°.	Tag pattern (English)	Tag pattern (French)
1	AN	NA
2	ANN	NNA
3	NN	NN
4	AAN	NAA
5	NAN	NNA
6	NPN	NPN
7	NNN	NNN
8	ANPN	NAPN
9	NPAN	NPNA
10	TN	TN

Table 3: Frequent Part-Of-Speech tag patterns used to transform syntactic structures of the sentence to translate from English to French. In these patterns A refers to an Adjective, P to a Preposition, T to Past Participle, and N to a Noun.

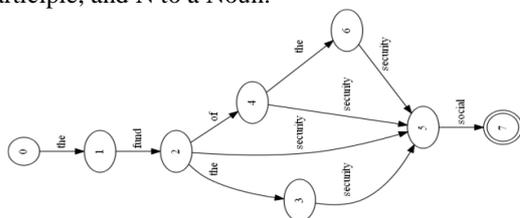


Figure 2: Example of a lattice of words corresponding to the syntactic transformation of the compound word “Social security funds”.

For example, from the sentence to translate “Social security funds in Greece encourage investment in innovation.”, two nominal chains are recognized: “Social security funds in Greece” and “investment in innovation”. These nominal chains are linked with the verb “encourage”. The expression “investment in innovation” is transformed using the sixth rule (Table 3) into the expression “the investment in the innovation”. It is important to mention here that the linking word “the” (definite article) is added to the applied rule before each noun (investment, innovation) in order to complete the transformation. The FSM transducer of the syntactic transfer step produces a lattice of words in the source language (Figure 2). Each word is represented with its lemma in the lattice and is associated with its linguistic information (Part-Of-Speech, gender, number, etc.).

Lexical transfer translates in the target language the lemmas of the obtained syntactic structures words using the bilingual lexicon of the cross-language search engine. This English-French lexicon is composed of 243539 entries. These entries are represented in their normalized forms (lemmas). A lemmatization process provided by the linguistic analyzer LIMA is applied on the obtained syntactic structures words. This step produces an important number of translation hypotheses. This is due to the combination of the syntactic transfer rules and the polysemy in the bilingual lexicon. The result of the bilingual reformulator is a set of lattices in which words are in the target language.

### 3.3 The Generator of Translations

The role of the generator of translations consists in assembling the results returned by the cross-language search engine and the bilingual reformulator, and in choosing the n-best translations according to a statistical language model learned from the target language corpus. The assembling process consists in composing FSMs corresponding to the translation candidates with FSMs corresponding to the translation hypotheses. The FSM state where the composition is made is determined by words which link the nominal chains of the translation candidates and the translation hypotheses. All the operations applied on the FSMs are made with the AT&T

FSM Library<sup>3</sup> (Mohri et al., 2002). In order to find the best translation hypothesis from the set of word lattices (Dong et al., 2014), a statistical model is learned with the CRF++ toolkit<sup>4</sup> (Lafferty et al., 2001) on lemmas and Part-Of-Speech tags of the target language corpus. Therefore, the n-best translations words are in their normalized forms (lemmas). To generate the n-best translations with words in their surface (inflected) forms, we applied a morphological generator (flexor) which uses the linguistic information (Part-Of-Speech, gender, number, etc.). The word lattices corresponding to the translations are enriched with the results of the flexor. These lattices are then scored with another statistical language model learned from texts of the target language containing words in inflected forms. The CRF++ toolkit is used to select the n-best translations in inflected forms.

## 4 Experiments and Results

### 4.1 Data and Experimental Setup

In order to study the impact of using a domain-specific bilingual lexicon on the performance of the EBMT system, we conducted our experiments on two English-French parallel corpora (Table 4): Europarl (European Parliament Proceedings) and Emea (European Medicines Agency Documents). Both corpora were extracted from the open parallel corpus OPUS (Tiedemann, 2012). Evaluation consists in comparing translation results produced by the open source SMT system Moses (Koehn et al., 2007) and the EBMT system on in-domain and out-of-domain texts. The English-French training corpus is used to build Moses’s translation and language models. The French sentences of this training corpus are used to create the indexed database of the cross-language search engine integrated in the EBMT system. We conducted eight runs and two test experiments for each run: In-Domain and Out-Of-Domain. For this, we randomly extracted 500 parallel sentences from Europarl as an In-Domain corpus and 500 pairs of sentences from Emea as an Out-Of-Domain corpus. These experiments are done to show the impact of the domain vocabulary on the translation results.

<sup>3</sup> FSM Library is available from AT&T for non-commercial use as executable binary programs.

<sup>4</sup> <http://wing.comp.nus.edu.sg/~forecite/services/parscit-100401/crfpp/CRF++0.51/doc/>.

The domain vocabulary is represented in the case of Moses by the specialized parallel corpus (Emea) which is added to the training data (Europarl). In the case of the EBMT system, the domain vocabulary is identified by a bilingual lexicon which is extracted automatically from the specialized parallel corpus (Emea) using a word alignment tool (Semmar et al., 2010; Bouamor et al., 2012). This specialized bilingual lexicon is added to the English-French lexicon which is used jointly by the cross-language search engine and the bilingual reformulator. To evaluate the performance of the EBMT system and Moses, we used the BLEU score (Papineni et al; 2002).

Run n°.	Training (# sentences)	Tuning (# sentences)
1	150K (Europarl)	3.75K (Europarl)
2	150K+10K (Europarl+Emea)	1.5K (Europarl)
3	150K+20K (Europarl+Emea)	1.5K (Europarl)
4	150K+30K (Europarl+Emea)	1.5K (Europarl)
5	500K (Europarl)	2.5K (Europarl)
6	500K+10K (Europarl+Emea)	2K+0.5K (Europarl+Emea)
7	500K+20K (Europarl+Emea)	2K+0.5K (Europarl+Emea)
8	500K+30K (Europarl+Emea)	2K+0.5K (Europarl+Emea)

Table 4: Corpora details used to train Moses language and translation models, and to build database of the EBMT system. In this table, K refers to 1000.

### 4.2 Results and Discussion

The performance of Moses and the EBMT system is evaluated using the BLEU score on the two test sets for the eight runs described in the previous section. Note that we consider one reference per sentence. The obtained results are reported in Table 5.

Run n°.	In-Domain		Out-Of-Domain	
	Moses	EBMT	Moses	EBMT
1	34.79	30.57	13.62	24.27
2	32.62	30.10	22.96	27.80
3	33.81	29.60	23.30	28.70
4	34.25	28.70	24.55	29.50
5	37.25	33.12	14.74	26.94
6	37.62	32.10	22.68	29.02
7	37.40	31.03	26.50	33.26
8	37.43	29.92	29.26	36.84

Table 5: BLEU scores of Moses and the EBMT system.

The first observation is that, when the test set is In-Domain, we achieve a relatively high BLEU score for both the two systems and the score of Moses is better in all the runs. For the Out-Of-Domain test corpus, the EBMT system performs better than Moses in all the runs and in particular Moses has obtained a very low BLEU score in the first and fifth runs (13.62 and 14.74). Furthermore, it seems that the English-French lexicon used in the cross-language search engine and the bilingual reformulator has had a significant impact on the result of the EBMT system. It improved regularly its BLEU score in all the runs. Likewise, these results show that small amounts of in-domain parallel data are more important for translation quality of Moses than large amounts of out-of-domain data. For example, adding a specialized parallel corpus composed of 30000 sentences to the 500000 sentences of Europarl reported a gain of 14.52 BLEU points. However, for the In-Domain test corpus, Moses’s BLEU score in runs 7 and 8 (adding respectively 20000 and 30000 sentences to the 500000 sentences of Europarl) is little than Moses’s BLEU score in run 6 (adding only 10000 sentences to the 500000 sentences of Europarl).

In order to evaluate qualitatively the EBMT system and Moses when translating specific and general-purpose texts, we take two examples of translations drawn from texts relating to the European Medicines Agency texts and the European Parliament proceedings (Tables 6 and 7). For the In-Domain sentence (Example 1), the EBMT system and Moses provide close translations and these translations are more or less correct. In the first example, the English word “keep” was identified by the morpho-syntactic analyzer used by the EBMT system as a verb and the bilingual lexicon proposed respectively the words “garder” and “continuer” as translations for this word. Of course, the translation proposed in the first run (garder) is correct but it is less expressive than the one proposed in the fifth run (continuer). The English-French lexicon proposes for the word “keep” several translations (continuer, entretenir, garder, maintenir, observer, protéger, respecter, tenir, etc.) but the EBMT system has chosen “garder” in run 1 and “continuer” in run 6. On the other hand, Moses added the preposition “de” (instead of the definite article “la”) to the word “cohésion” when it translated the word

“cohesion” in the expression “solidarity and cohesion”.

<b>Example 1 Input:</b> our success must be measured by our capacity to <i>keep</i> growing while ensuring <i>solidarity and cohesion</i> .	
<b>Reference</b>	nous devons mesurer notre réussite à notre capacité à <i>poursuivre sur la voie</i> de la croissance tout en garantissant <i>la solidarité et la cohésion</i> .
<b>EBMT system: Run 1</b>	notre succès doit être mesuré à notre capacité à <i>garder</i> la croissance en garantissant <i>la solidarité et la cohésion</i> .
<b>EBMT system: Run 6</b>	notre succès doit être mesuré à notre capacité à <i>continuer</i> la croissance en garantissant <i>la solidarité et la cohésion</i> .
<b>Moses: Run 1</b>	notre succès doit être mesuré par notre capacité à <i>maintenir</i> la croissance tout en assurant <i>la solidarité et de cohésion</i> .
<b>Moses: Run 6</b>	notre succès doit être mesuré par notre capacité à <i>suivre</i> la croissance, tout en assurant <i>la solidarité et de cohésion</i> .

Table 6: Translations produced by the EBMT system and Moses for an In-Domain sentence.

For the Out-Of-Domain sentence (Example 2), the EBMT system results are clearly better and most of the translations produced by Moses are incomprehensible and ungrammatical. This result can be explained by the fact that the test corpus has a vocabulary which is different from the entries of Moses’s translation table. For instance, the EBMT system translates correctly the compound words “fasting blood glucose” and “total cholesterol” (glycémie à jeun, cholestérol total) but it translates the compound word “routine care group” as “groupe de soins de routine” instead of “groupe de soins routiniers”. As we can see, this translation could not be provided by the bilingual reformulator because there is no transfer rule implementing the tag pattern of this compound word which is NPNP (Table 3). This expression corresponds to a partial translation provided by the cross-language search engine for the sentence to translate. On the other hand, Moses fails to translate correctly the multiword expressions “fasting blood glucose”, “total cholesterol”, “duloxetine-treated patients” and “routine care group” in run 4. However, it succeeds in the translation of the expressions “fasting blood glucose” and “total cholesterol” in run 8.

<b>Example 2 Input:</b>	there was also a small increase in <i>fasting blood glucose</i> and in <i>total cholesterol</i> in duloxetine-treated patients while those laboratory tests showed a slight decrease in the <i>routine care group</i> .
<b>Reference</b>	il y a eu également une faible augmentation de la <i>glycémie à jeun</i> et du <i>cholestérol total</i> dans le groupe duloxétine alors que les tests en laboratoire montrent une légère diminution de ces paramètres dans le <i>groupe traitement usuel</i> .
<b>EBMT System: Run 4</b>	il y avait aussi une petite augmentation dans la <i>glycémie à jeun</i> et du <i>cholestérol total</i> chez les patients traités par la duloxétine alors que les tests en laboratoire montraient une légère diminution dans le <i>groupe de soins de routine</i> .
<b>EBMT System: Run 8</b>	il y avait aussi une faible augmentation dans la <i>glycémie à jeun</i> et du <i>cholestérol total</i> chez les patients traités par la duloxétine alors que les tests en laboratoire montraient une légère diminution dans le <i>groupe de soins de routine</i> .
<b>Moses: Run 4</b>	il était également une légère augmentation de répréhensible <i>glycémie artérielle</i> et en total de patients duloxetine-treated <i>cholesterol</i> laboratoire alors que ces tests, ont montré une diminution sensible dans les <i>soins standards groupe</i> .
<b>Moses: Run 8</b>	il y a aussi une légère augmentation de la <i>glycémie à jeun</i> et <i>cholestérol total</i> de patients duloxetine-treated alors que ces tests de laboratoire a montré une légère baisse dans les <i>soins de routine groupe</i> .

Table 7: Translations produced by the EBMT system and Moses for an Out-Of-Domain sentence.

After analyzing some translations, we observed that the major issues of our EBMT system are related to errors from the source-language syntactic analyzer, the non-isomorphism between the syntax of the two languages and the polysemy in the bilingual lexicon. To handle the first two issues, we proposed to take into account translation candidates returned by the cross-language search engine even if these translations correspond only to a part of the sentence to translate. However, for the presence of the polysemy in the bilingual lexicon, the EBMT system has no specific treatment. This can explain partially why the EBMT system is outperformed by Moses when translating In-Domain sentences. It seems that translation table probabilities which are computed during the word alignment process with Giza++ (Och and Ney,

2002; Och and Ney, 2003) have contributed to choose the right translation. On the other hand, we noted that most of Moses’s translation errors for Out-Of-Domain sentences are related to vocabulary. For example, Moses proposes the compound word “glycémie artérielle” as a translation for the expression “fasting blood glucose” in run 4 which is not correct. In SMT systems such as Moses, phrase tables are the main knowledge source for the machine translation decoder. The decoder consults these tables to figure out how to translate an input sentence from the source language into the target language. These tables are built automatically using the open source word alignment tool Giza++. However, Giza++ could produce errors in particular when it aligns multiword expressions. (Bouamor et al, 2012; Ren et al., 2009) showed that the integration of multiword expressions in Moses’s translation model improves the translation quality. Multiword expressions include a large list of categories such as collocations, compound words, idiomatic expressions, named entities and domain-specific terms (Baldwin and Kim, 2010). To reduce word alignment errors with Giza++, we propose the following three methods to integrate into Moses the bilingual lexicon which is extracted automatically by our word alignment tool from the specialized parallel corpus (Emea):

- **Moses<sub>CORPUS</sub>**: In this method, we add the extracted bilingual lexicon as a parallel corpus and retrain the translation model. By increasing the occurrences of the specialized words and their translations, we expect a modification of alignment and probability estimation.
- **Moses<sub>TABLE</sub>**: This method consists in adding the extracted bilingual lexicon into Moses’s phrase table. We use the *Cosine* similarity measure provided by our word alignment tool for each specialized word of the bilingual lexicon as a translation probability.
- **Moses<sub>FEATURE</sub>**: In this method, we extend “Moses<sub>TABLE</sub>” by adding a new feature indicating whether a word comes from the specialized bilingual lexicon or not (1 or 0 is introduced for each entry of the phrase table).

For these experiments, we used only English-French training corpora of runs 2, 3 and 4 to build Moses’s translation and language models. We measure the translation quality on the same test sets of the previous experiments (500 parallel sentences extracted randomly from Europarl for the In-Domain test and 500 pairs of sentences extracted randomly from Emea for the Out-Of-Domain test). Because the bilingual lexicon which is extracted automatically from the specialized parallel corpus is composed of entries in their normalized forms (lemmas), we used the factored translation model of Moses (Koehn et al., 2010). This model accepts the use of additional annotations at the word level and operates on lemmas instead of surface forms. The translation process consists, first, in translating lemmas of words from the source language into the target language, and second, in generating the inflected forms for each lemma. Tables 8 and 9 present respectively the Moses’s results for the In-Domain and the Out-Of-Domain sentences when using the three integration strategies.

The first important point to mention here is that there is improvement of the BLEU score in all the integration methods for the Out-Of-Domain sentences. The best improvement is achieved using the  $Moses_{FEATURE}$  method which guides Moses to choose specialized words instead of those provided by the translation model built with Giza++. Compared to the baseline system (Moses without using integration strategies), this method reports a gain of 3.63 BLEU points for the fourth run. The obtained BLEU score (28.18) is not very far from the BLEU score obtained by the EMBT system in the same run (29.50). We think that this high score is due to the feature which guides Moses in choosing the best translation with a preference to the words of the specialized bilingual lexicon. In this case, Moses neglects the other translations found in the translation table. On the other hand, the  $Moses_{TABLE}$  method has lower scores in all the runs. We assume that we obtain such lower scores because the content of the translation table is not coherent. Indeed, we considered the *Cosine* similarity measure provided by our word alignment tool for each specialized word of the bilingual lexicon as a translation probability. However, in actual fact, values of *Cosine* similarity measures are not similar to translation probabilities provided by Giza++.

Run n°.	In-Domain		
	$Moses_{CORPUS}$	$Moses_{TABLE}$	$Moses_{FEATURE}$
2	32.82	32.15	29.18
3	33.89	33.48	30.26
4	34.64	34.11	31.84

Table 8. Translation results in terms of BLEU scores corresponding to the three integration methods for the In-Domain sentences.

Run n°.	Out-Of-Domain		
	$Moses_{CORPUS}$	$Moses_{TABLE}$	$Moses_{FEATURE}$
2	23.45	23.11	24.69
3	24.09	23.76	25.68
4	25.43	25.05	28.18

Table 9. Translation results in terms of BLEU scores corresponding to the three integration methods for the Out-Of-Domain sentences.

As it can be seen, these results confirm that adding specialized parallel corpora to the training data improves the translation quality of Out-Of-Domain test corpus for the both MT systems in all cases but the improvement of the EBMT system is more significant. Likewise, even if the size of the specialized corpus and the size of the general-purpose monolingual corpus are not significant, the EBMT prototype produces correct translations for both in-domain and out-of-domain texts.

## 5 Conclusion

In this paper, we have studied the impact of using a domain-specific corpus on the performance of an EBMT system and Moses. Two kinds of texts are used in our experiments: in-domain texts from Europarl and out-of-domain texts from Emea. We have seen that both the two systems achieved a relatively high BLEU score for in-domain texts. Our experiments on out-of-domain texts have showed that the EBMT system performs better than Moses. Moreover, we have noticed that the method which guides Moses to choose specialized words instead of those provided by the translation model built with Giza++ achieves the best improvement. In the future, we plan, on the one hand, to use machine learning techniques to extract transfer rules for the bilingual reformulator from annotated parallel corpora, and on the other hand, to evaluate the EBMT system on other specific domains such as security, finance, etc.

## References

- Ricardo Baeza-Yates and Berthier Ribeiro-Neto. 1999. Modern Information Retrieval. In *Addison-Wesley Longman Publishing Co., Inc.*
- Timothy Baldwin and Su Nam Kim. 2010. Multiword Expressions. In *Indurkha and Damerau Handbook of Natural Language Processing, Second Edition*, pages 267-292.
- Pratyush Banerjee, Jinhua Du, Baoli Li, Sudip Kr. Naskar, Andy Way, and Josef van Genabith. 2010. Combining Multi-Domain Statistical Machine Translation Models using Automatic Classifiers. In *Proceedings of the Ninth Conference of the Association for MT in the Americas*, pages 141–150.
- Nicola Bertoldi and Marcello Federico. 2009. Domain adaptation for statistical machine translation with monolingual resources. In *Proceedings of the 4<sup>th</sup> Workshop on Statistical Machine Translation*.
- Romaric Besançon, Gaël De Chalendar, Olivier Ferret, Christian Fluhr, Olivier Mesnard, and Hubert Naets. 2003. Concept-Based Searching and Merging for Multilingual Information Retrieval: First Experiments at CLEF 2003. In *C. Peters et al. (Ed.): CLEF 2003, Springer Verlag, Berlin*.
- Romaric Besançon, Gaël De Chalendar, Olivier Ferret, Faïza Gara, Meriama Laib, Olivier Mesnard, and Nasredine Semmar. 2010. LIMA: A multilingual framework for linguistic analysis and linguistic resources development and evaluation. In *Proceedings of the seventh international conference on Language Resources and Evaluation*.
- Dhouha Bouamor, Nasredine Semmar, and Pierre Zweigenbaum. 2012. Automatic Construction of a Multiword Expressions Bilingual Lexicon: A Statistical Machine Translation Evaluation Perspective. In *Proceedings of the 3<sup>rd</sup> Workshop on Cognitive Aspects of the Lexicon, COLING 2012*.
- Dhouha Bouamor, Nasredine Semmar, and Pierre Zweigenbaum. 2012. Identifying bilingual multiword expressions for statistical machine translation. In *Proceedings of the 8<sup>th</sup> international conference on Language Resources and Evaluation*, pages 674-679, Turkey.
- Lars Bungum and Bjorn Gambäck. 2011. A Survey of Domain Adaptation in Machine Translation Towards a refinement of domain space. In *Proceedings of the India-Norway Workshop on Web Concepts and Technologies*.
- Jorge Civera and Alfons Juan. 2007. Domain adaptation in statistical machine translation with mixture modelling. In *Proceedings of the Second Workshop on Statistical Machine Translation*.
- Hal Daumé III and Jagadeesh Jagarlamudi. 2011. Domain Adaptation for Machine Translation by Mining Unseen Words. In *Proceedings of the 49<sup>th</sup> Annual Meeting of the Association for Computational Linguistics: short papers*, pages 407–412.
- Mark W. Davis and William C. Ogden. 1997. QUILT: Implementing a large-scale cross-language text retrieval system. In *Proceedings of SIGIR*.
- Meiping Dong, Yong Cheng, Yang Liu, Jia Xu, Maosong Sun, Tatsuya Izuha, and Jie Hao. 2014. Query Lattice for Translation Retrieval. In *Proceedings of COLING 2014, the 25<sup>th</sup> International Conference on Computational Linguistics: Technical Papers*, pages 2031–2041, Dublin, Ireland.
- Gregory Grefenstette. 1998. Cross-Language Information Retrieval. In *The Information Retrieval Series, Vol. 2, Springer*.
- Almut Silja Hildebrand, Matthias Eck, Stephan Vogel, and Waibel Alex. 2005. Adaptation of the translation model for statistical machine translation based on information retrieval. In *Proceedings of the European Association for Machine Translation Conference*.
- John Hutchins. 2003. Machine Translation: General Overview. In *Ruslan (Ed.), The Oxford Handbook of Computational Linguistics, Oxford: University Press*, pages 501-511.
- Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicolas Bertoldi, Brooke Cowan, Wade Shen, Christine Mo-ran, Richard Zens, Chris Dyer, Ondrej Bojar, Al-exandra Constantin, and Evan Herbst. 2007. Moses: Open source toolkit for statistical machine translation. In *Proceedings of the Conference ACL 2007, demo session, Prague, Czech Republic*.
- Philipp Koehn, Barry Haddow, Philip Williams, and Hieu Hoang. 2010. More Linguistic Annotation for Statistical Machine Translation. In *Proceedings of the Fifth Workshop on Statistical Machine Translation and Metrics*.
- John Lafferty, Andrew McCallum, and Fernando C. N. Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the Eighteenth International Conference on Machine Learning*.

- Philippe Langlais. 2002. Improving a general-purpose statistical translation engine by terminological lexicons. In *Proceedings of COLING: Second international workshop on computational terminology*.
- William D. Lewis, Chris Wendt, and David Bullock. 2010. Achieving Domain Specificity in SMT without Overt Siloing. In *Proceedings of the seventh international conference on Language Resources and Evaluation*.
- Mehryar Mohri, Fernando Pereira, and Michael Riley. 2002. Weighted Finite-State Transducers in Speech Recognition. In *Computer Speech and Language*, 16(1), pages 69-88.
- Franz Josef Och and Hermann Ney. 2002. Discriminative training and maximum entropy models for statistical machine translation. In *Proceedings of the 40<sup>th</sup> meeting of the Association for Computational Linguistics*.
- Franz Josef Och and Hermann Ney. 2003. A Systematic Comparison of Various Statistical Alignment Models. In *Computational Linguistics*, volume 29, number 1, pages 19-51.
- Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a method for automatic evaluation of machine translation. In *Proceedings of the 40<sup>th</sup> Annual meeting of the Association for Computational Linguistics*, pages 311-318.
- Pavel Pecina, Antonio Toral, Andy Way, Vassilis Papavassiliou, Prokopis Prokopidis, and Maria Giagkou. 2011. Towards Using Web-Crawled Data for Domain Adaptation in Statistical Machine Translation. In *Proceedings of the 15<sup>th</sup> Conference of the European Association for Machine Translation*.
- Zhixiang Ren, Yajuan Lu, Jie Cao, Qun Liu, and Yun Huang. 2009. Improving statistical machine translation using domain bilingual multiword expressions. In *Proceedings of the Workshop on Multiword Expressions, ACL-IJCNLP*, pages 47-57, Suntec, Singapore.
- Gerard Salton and Michael J. McGill. 1986. Introduction to Modern Information Retrieval. In *McGraw-Hill, Inc.*
- Nasredine Semmar, Meriama Laib and Christian Fluhr. 2006. A Deep Linguistic Analysis for Cross-language Information Retrieval. In *Proceedings of the International Conference on Language Resources and Evaluation, LREC*, Italy.
- Nasredine Semmar and Meriama Laib. 2010b. Using a Hybrid Word Alignment Approach for Automatic Construction and Updating of Arabic to French Lexicons. In *Proceedings of the Workshop on LR and HLT for Semitic Languages, LREC*.
- Nasredine Semmar, Dhouha Bouamor. 2011. A New Hybrid Machine Translation Approach Using Cross-Language Information Retrieval and Only Target Text Corpora. In *Proceedings of the International Workshop on Using Linguistic Information for Hybrid Machine Translation, Spain*.
- Nasredine Semmar, Christophe Servan, Dhouha Bouamor, and Ali Joua. 2011. Using Cross-Language Information Retrieval for Machine Translation. In *Proceedings of the 5<sup>th</sup> Language & Technology Conference, Poland*.
- Nasredine Semmar, Othman Zennaki, and Meriama Laib. 2015. Evaluating the Impact of Using a Domain-specific Bilingual Lexicon on the Performance of a Hybrid Machine Translation Approach. In *Proceedings of the 10<sup>th</sup> International Conference on Recent Advances in Natural Language Processing, Bulgaria*.
- Matthew Snover, Bonnie Dorr, and Richard Schwartz. 2008. Language and translation model adaptation using comparable corpora. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*.
- Ankit Srivastava, Sergio Penkale, Declan Groves, and John Tinsley. 2009. Evaluating Syntax-Driven Approaches to Phrase Extraction for MT. In *Proceedings of the 3<sup>rd</sup> International Workshop on Example-Based Machine Translation, Ireland*.
- Jörg Tiedemann. 2012. Parallel Data, Tools and Interfaces in OPUS. In *Proceedings of the 8<sup>th</sup> International Conference on Language Resources and Evaluation*.
- Arturo Trujillo. 1999. Translation Engines: Techniques for Machine Translation. In *Applied Computing, Springer*.
- Veronis Jean, Hamon Olivier, Ayache Christelle, Belmouhoub Rachid, Kraif Olivier, Laurent Dominique, Nguyen Thi Minh Huyen, Semmar Nasredine, Stuck François and Wajdi Zaghouni. 2008. L'évaluation des technologies de traitement de la langue: La campagne d'évaluation Arcade II. In *Chapitre 2, Editions Hermès, 2008*.
- Wei Wang, Klaus Macherey, Wolfgang Macherey, Franz Och, and Peng Xu. 2012. Improved Domain Adaptation for Statistical Machine Translation. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*.

# A Multifactorial Analysis of English Particle Movement in Korean EFL Learners' Writings

**Gyu-Hyeong Lee**

Hannam University  
70 Hannamro, Daedeok-gu  
Daejeon 306-791, Korea  
gyuhyung73@naver.com

**Ha-Eung Kim**

Hannam University  
70 Hannamro, Daedeok-gu  
Daejeon 306-791, Korea  
tankkh@hanmail.net

**Yong-hun Lee**

Chungnam Nat'l University  
99 Daehak-ro, Yuseong-gu  
Daejeon 305-764, Korea  
yleeuiuc@hanmail.net

## Abstract

This paper investigated Particle Movement in Korean EFL learners' writings. Gries (1999, 2001, 2003) adopted a multifactorial analysis to examine Particle Movement of native speakers. Several linguistics factors were proposed in the studies, and it was demonstrated that these factors influenced the choice of the constructions. This paper also employed a multifactorial analysis to examine Particle Movement in Korean EFL learners' writings. The analysis results illustrated that the Korean EFL learners were slightly different from native speakers in that only some factors were used for the selection of constructions.

## 1 Introduction

Linguistic alternation is one of the interesting areas in the linguistic investigations. Particle Movement is one of the syntactic alternations. It refers to the phenomenon where a particle goes behind the direct object (DO) in the phrasal verb constructions. Let's see the following example (Gries, 1999:1).

- (1) a. John *picked up* the book.  
b. John *picked* the book *up*.

In (1a), the order is 'verb + particle + DO'. However, the particle is separated from the verb

in (1b). That is, (1b) has an order of 'verb + DO + particle'.

There have been a lot of theoretical studies to investigate what linguistic factors determine the choice of the alternation, in traditional grammar and generative grammar. Nowadays, as computer technology and statistics develop, there have been a few studies to explain these syntactic phenomena with corpus data. Gries (1999, 2001, 2003) adopted a multifactorial analysis to examine the Particle Movement in the native speakers' writings. These studies proposed several linguistics factors and it was demonstrated that these factors and their interactions significantly influenced the choice of the constructions

This paper also adopted a multifactorial analysis to examine the Particle Movements in Korean EFL learners' writings. The Korean part of TOEFL11 corpus was used, and all the relevant sentences were extracted using the C7 tag information. The relevant factors were encoded to these sentences, and each factor was statistically analyzed with R. Through the analysis, it was demonstrated that Korean EFL learners employed a different strategy in Particle Movement and that only some factors were used for the selection of alternation.

This paper is organized as follows. In Section 2, previous studies are reviewed with focused on corpus-based approaches. Section 3 enumerates research methods, and Section 4 contains analyses results. Section 5 is for discussions, and Section 6 summarizes this paper.

## 2 Previous Studies

### 2.1 On Particle Movement

There have been several studies on English Particle Movement in various linguistic fields: traditional grammar (Sweet, 1892; Jespersen, 1928; Krusinga and Erades, 1953), Chomskyan transformational-generative grammar (Fraser, 1974, 1976; Den Dikken, 1992, 1995; Rohrbacher, 1994), cognitive grammar (Yeagle, 1983), discourse-functional approaches (Chen, 1986), psycholinguistically-oriented approaches (Hawkins, 1994), and so on.

Gries (1999:33) closely investigated the claims in previous studies and summarized them as follows.

Value for construction <sub>0</sub>	Variable	Value for construction <sub>1</sub>
Long DO	Length of the DO in words (Length W)	
Long DO	Length of the DO in syllables (LengthS)	
Complex	Complexity of the DO (Complex)	
	NP-Type of the DO: semi-pronominal (Type)	pronominal
Indefinite	Determiner of the DO (Det)	definite
No	Previous mention of the DO (Lm)	yes
Low	← Times of preceding mention of the DO (Topm) →	high
High	← Distance to last mention of the DO (Dltn/ActPC) →	low
High	← News Value of the DO →	low
Yes	(Contrastive) Stress of the DO	
Yes	Subsequent mention of the DO (Nm)	no
High	← Times of subsequent mention of the DO (Tosm) →	low
How	← Distance to next mention of the DO (Dtnm/ClusSC) →	high
	Overall frequency of the DO (O <sub>tot</sub> )	
	following directional adverbial (PP)	yes
Yes	Prep of the following PP is identical to the particle (Part = Prep)	
	Register	
Idiomatic	← Meaning of the VP (Idiomacity) →	literal
Low	← Cognitive Entrenchment of the DO →	high
Inanimate	Animacy of the DO (Animacy)	animate
Abstract	Concreteness of the DO (Concreteness)	concrete

**Table 1.** Variables That Govern the Alternation

Here, *construction<sub>0</sub>* refers to the sentences with the order of ‘verb + particle + DO’ as in (1a), while *construction<sub>1</sub>* refers to the sentences with the order of ‘verb + DO + particle’ as in (1b). This table enumerated 18 different linguistic factors and demonstrated that several different types of factors, not a single factor, actually influenced the choice of the constructions.

Let's see how these factors can be related with the alternation of Particle Movement. For example, LENGTHW (the first factor in Table 1) refer to the length of DO in words. If the DO is long, native speakers tend to choose *construction<sub>0</sub>* rather than *construction<sub>1</sub>*. If the DO is short, the native speakers prefer *construction<sub>1</sub>* to *construction<sub>0</sub>*. The factor DET, the fifth factor, refers to the determiner of the DO. If the determiner of DO is indefinite (such as *a* or *an*), native speakers tend to choose *construction<sub>0</sub>* rather than *construction<sub>1</sub>*. If the determiner of

DO is definite (such as *the*), native speakers prefer *construction<sub>1</sub>* rather than *construction<sub>0</sub>*. Table 1 contains all the related factors which cover most of linguistic fields: phonology, syntax, semantics, pragmatics, and discourse analysis.

### 2.2 Corpus-based Studies

Although it is fact that previous studies contributed to find out linguistic factors influenced the choice of alternation, their data exclusively relied on native speakers' intuition. Gries (2001, 2003) pointed out this problem and performed an analysis based on the corpus data.

Gries (2001:36-37) pointed out three problems of these previous approaches. First, most variables were based on introspective analysis and non-authentic example sentences. This problem is due to the fact that previous studies exclusively relied on the native speakers' intuition (viz. acceptability judgments). The problems of this type of test are (i) that they do not necessarily constitute objective, reliable, and valid data, (ii) that it is questionable that an analysis based on these data can in fact produce representative results, and (iii) that it is possible to evaluate sentences produced artificially, out of context. Second, most previous analyses only performed the monofactorial analyses, where only one variable has an effect on the alternation in isolation. The problem of monofactorial analysis is that the examples do not warrant the claim that the preference for one construction over the other need not be related the relevant factor exclusively. Instead, the tendency might come from other factors. Therefore, given that factors are encoded in the determination of the constructions, it is difficult to solely rely on monofactorial analyses to describe particle movement adequately. Third, there have been only a few analyses aiming at subsuming all the variables under a common basis and there has been no analysis has aimed at predicting particle placement in natural discourse situations.

In order to solve this problem, Gries (2001, 2003) employed a multifactorial analysis, where all the factors in Table 1 were taken into consideration simultaneously. These studies used a Generalized Linear Model (GLM) and statistically analyzed how each factor played a role in the choice of construction. They also took a linear discriminant analysis (LDA) and a classification and regression tree (CART) and calculated the importance of each factor as follows (Gries, 2001:48).

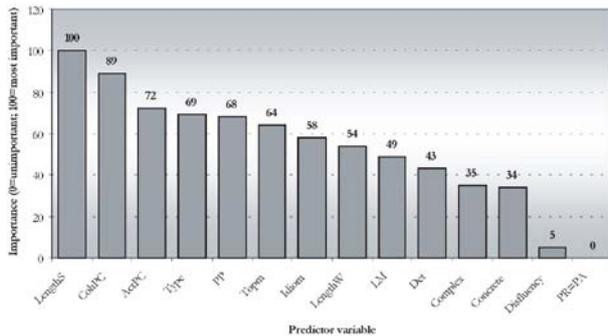


Figure 1. Importance of Predictors for CART

As this figure indicates, not all the linguistic factors play roles in the choice of alternation. In addition, some factors are more important, and others are less important.

Gries (2001) and Gries (2003) were essentially different from the previous approaches, since (i) these studies made use of corpus data (naturally occurring data) and (ii) they statistically analyzed the collected data.

### 3 Research Method

#### 3.1 Questions and Hypothesis

Although there have been a lot of studies on Particle Movement in native speakers, there are few studies on the phenomena of the EFL learners. This study investigated the Particle Movement of Korean EFL learners.

Through the analysis, this paper wants to answer the following research questions.

- (2) a. Do Korean EFL learners show the same or similar tendency in Particle Movement in their writings?
- b. If Korean EFL learners employ different factors, which factors were employed in their choice of alternation?
- c. Does the ratio of these two constructions (*construction<sub>0</sub>* vs. *construction<sub>1</sub>*) change as the level of proficiency goes up?

For these research questions, the following hypothesis was made.

- (3) a. If Korean EFL learners show the same or similar tendency that native speakers demonstrate in their writings, two groups of people may employ similar factors or a similar set of factors in their writings that influence the choice of constructions.

- b. If Korean EFL learners show a different tendency from the native speakers, two groups may employ different factors or a different combinations of factors in their writings which decide the choice of constructions.

In order to answer these questions, the following investigations were conducted.

#### 3.2 Corpus

This study employed two types of data. The first one was the TOEFL11 corpus for the EFL learners (LDC Catalo No.: LDC2014T06), and the second one was the data in Gries (2001, 2003) for the native speakers (as reference data set). The second data were not the actual data but the analysis results in Gries (2001, 2003).

The TOEFL11 corpus was released by the English Testing Service (ETS) in 2014. The corpus consists of essays written during the TOEFL iBT® tests in 2006-2007 (Blanchard et al., 2013). It contains 1,100 essays per each of the 11 native languages, totaling 12,100 essays. All of the essays were taken from the TOEFL independent task, where test-takers were asked to write an essay in response to a brief writing topic. The essays were sampled as evenly as possible from eight different topics. The corpus also provides the score levels (Low/Medium/High) for each essay.

There are other kinds of corpora which can be used for examining the use of EFL learners. The International Corpus of Learner English (ICLE; Granger et al., 2009) is one example. Although the ICLE is a good resource to explore linguistic properties of EFL learners' use of English, there are several reasons for choosing the TOEFL11 instead of the ICLE.

First, the TOEFL11 corpus includes essays written by Korean EFL learners, while the ICLE does not.<sup>1</sup>

Second, each essay in the TOEFL11 corpus contains information on score levels. The score levels were calculated first by combining the individual 5-point-scale scores given by the human raters and then by collapsing this

<sup>1</sup> The ICLE corpus contains 16 components (Bulgarian, Chinese, Czech, Dutch, Finnish, French, German, Italian, Japanese, Norwegian, Polish, Russian, Spanish, Swedish, Tswana, and Turkish) but the TOEFL11 includes 11 L1s (French, Italian, Spanish, German, Hindi, Japanese, Korean, Turkish, Chinese, Arabic, and Telugu).

combined score into 3 levels (Low/Medium/High). The 5-point-scale human scores were determined by the defined criteria.

Third, the number of essays that the former corpus contains is bigger than that of the latter. The ICLE includes 380 essays per L1 (=6,085/16), while the TOEFL11 has 1,100 essays per L1. Thus, the TOEFL 11 contains about three times as many essays as the ICLE.

Fourth, one of the biggest problems of the ICLE is that essay topics are not evenly distributed across the 16 L1s. The language usage is heavily driven by a given essay topic. This implies that some of the ICLE data may be conflated by the uneven distribution of essay topics across the 16 L1s. It is important to investigate the linguistic properties of EFL learners in an evenly distributed corpus.

Finally, because of the differences in the essay tasks administered and responses collected, there were differences not only in character encodings but also corpus annotations across L1s. These differences make it difficult for the findings of one L1 to be generalized to other contexts.

### 3.3 Procedure

The analysis in this paper proceeded as follows.

First, all the writing samples of Korean EFL learners were extracted from the TOEFL 11 corpus. A total of 328,384 word tokens were included in the extracted corpus.

Second, the writing samples were classified into three levels. Through the classification, each level had the following corpus size (word token): 95,066 (High), 202,531 (Medium), and 30,787 (Low).

Third, each text was POS tagged with the C7 CLAWS taggers.<sup>2</sup>

Fourth, all the sentences with particles were extracted using NLPTools (Lee, 2007).<sup>3</sup>

Fifth, all the relevant factors were encoded to each sentence. This paper adopted 8 factors and they are enumerated in Table 2.

Among these, the first factor was newly introduced in this analysis and the others came from Table 1.

<sup>2</sup> You can easily use Free CLAWS WWW tagger in <http://ucrel.lancs.ac.uk/claws/trial.html>. For details of C7 tag sets, see Jurafsky and Martin (2009).

<sup>3</sup> In the C7 tag sets, particles have a tag RP. The reason why NLPTools was used here is that the software had a function which could extract the whole sentences with the given tag(s) (i.e., \*\_RP).

Finally, all the data were statistically analyzed using R.

Variable	Explanation
LEVEL	Level of proficiency
COMPLEXITY	Complexity of Direct Object
ANIMACY	Animacy of Direct Object
DEFINITENESS	Definiteness of Direct Object
PRONOMINALITY	Pronominality of Direct Object
IDIOMACITY	Idiomacity of Direct Object
CONCRATENESS	Concreteness of Direct Object
LENGTH	Length of Direct Object in Words

Table 2. Variables Used in the Analysis

## 4 Analysis Results

### 4.1 Descriptive Statistics

After all the sentences with the particles were extracted, the sentences were classified into two groups, based on the transitive vs. intransitive use of phrasal verbs. This process was necessary since the Particle Movement occurred in the transitive use of phrasal verb constructions.

The following graph illustrates the ratio of each group (intransitive vs. transitive) of phrasal verb constructions in Korean speakers' writings.

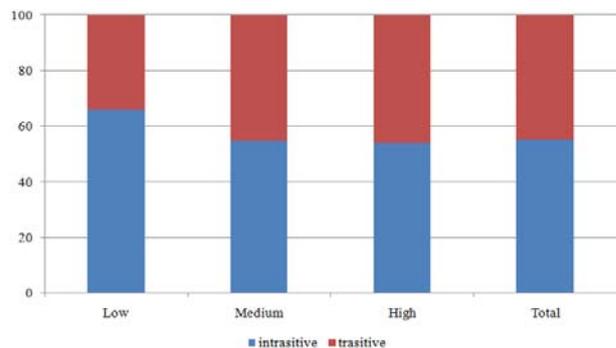


Figure 2. Intransitive vs. Transitive

Here, the lower part corresponds to the intransitive uses and the upper part to the transitive uses.

As this graph illustrated, Korean EFL learners preferred intransitive uses of phrasal verbs rather than the transitive uses. Note that nearly 50%-70% of sentences were intransitive uses of phrasal verbs. This tendency appeared in all the levels of proficiency, though the proportion of intransitive uses of phrasal verbs decreased as the level of proficiency went up. However, the proportion of the Medium level was indistinguishable from that of the High level.

Then, among the sentences with phrasal verbs, all the constructions which had transitive uses

were extracted, and the ratios of two constructions were calculated. Figure 3 illustrated the analysis results.

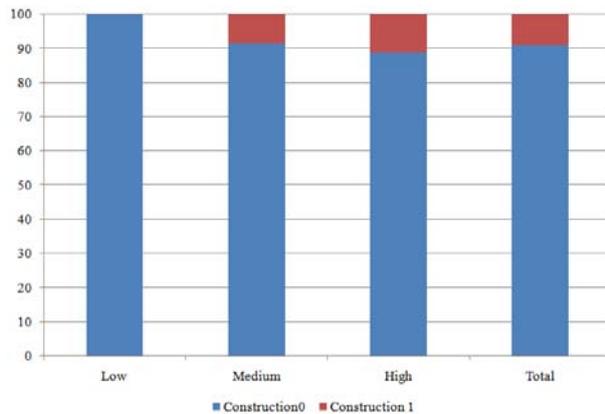


Figure 3. construction<sub>0</sub> vs. construction<sub>1</sub>

As this graph shows, the ratio of two constructions differ as the level goes up. The Korean EFL learners in the Low level used only construction<sub>0</sub>. In the Medium level, the proportion of construction<sub>0</sub> decreased and the Korean EFL learners began to use construction<sub>1</sub>. In the High level, the proportion of construction<sub>0</sub> decreased more, the proportion of construction<sub>1</sub> increased more. Overall, the Korean EFL learners preferred construction<sub>0</sub> to construction<sub>1</sub>.

#### 4.2 Inferential Statistics

From the data described in Figure 2, the sentences with transitive uses of phrasal verbs were extracted, since those sentences could be classified into one of the two constructions (either construction<sub>0</sub> or construction<sub>1</sub>). Then, a GLM was applied to the data, as in Gries (2001, 2003).

This model was chosen through the following steps. First, since we had 8 factors, a (Multiple) Linear Regression analysis is adopted (a multi-factorial analysis; Gries, 2003). Second, since the dependent variable CONSTRUCTION was binomial, a Generalized Linear Regression Model had to be used with logistic regression.

The initial model was constructed as follows.

- (4) Initial Model (Unsaturated)  
 $CONSTRUCTION \sim LEVEL + COMPLEXITY + ANIMACY + DEFINITENESS + PRONOMINALITY + IDIOMACITY + CONCRETENESS + LENGTH$

This is the initial model, where no interaction was included.

Then, a model selection process was performed. According to Gries (2013), there are two types of model selection parameters. One is based on the direction of the analysis and the other is the criterion determining whether or not a predictor gets to be in the model. On the direction of the analysis, most analyses have adopted a backward selection, and this paper also took this method. There are two types of approaches to the selection of relevant models: significance-based approaches and criterion-based approaches. This paper took a significance-based approach. That is, the analysis would start from the maximally saturated model, and continued to remove predictors (backward) until the analysis reached the statistically significant differences in the *p*-value (significant-based).

Since this paper adopted a backward selection, the first thing is to make a saturated model. The following model is a saturated model.

- (5) Saturated Model  
 $CONSTRUCTION \sim LEVEL * COMPLEXITY * ANIMACY * DEFINITENESS * PRONOMINALITY * IDIOMACITY * CONCRETENESS * LENGTH$

Note that all the interactions were included in this model.

Now that a saturated model was obtained, the statistical analysis started from the model. A new model was made by deleting one interaction or one factor from the saturated model. Then, it was checked whether this new model is significantly different from the previous model. If  $p < .05$ , it means that two models were significantly different and that the deleted factor or interaction MUST NOT be deleted from the model. If  $.05 < p$ , it means that two models were not significantly different and that the deleted factor or interaction can be deleted safely without distorting the explanatory power of the model. The selection procedures were continued until no redundant factor or interaction remained in the model. Through this process, the final model was obtained.

In the final model, there were lots of interactions among the factors. Since it is impossible and unreasonable to examine all the factors and their interactions, this paper examines only the effects of major factors. The following table contains the statistical values for each factor.

	Estimate	sd	z	p
(Intercept)	0.702	0.611	1.150	.250
LEVEL1	0.086	0.149	0.576	.565
LEVEL2	-0.213	0.228	-0.936	.349
COMPLEXITY1	1.757	0.520	3.376	<.001
ANIMACY1	-1.090	0.114	-9.578	<.001
DEFINITENESS1	0.194	0.102	1.896	.058
PRONOMINALITY1	-0.542	0.639	-0.849	.396
IDIOMACITY1	0.132	0.087	1.508	.132
CONCRETENESS1	1.127	0.439	2.265	.010
LENGTH	0.268	0.088	3.062	.002

**Table 3.** Analysis Results

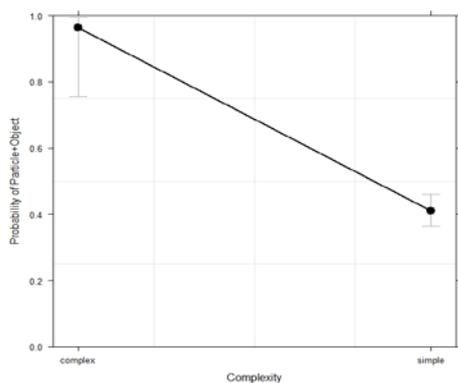
As this model shows, only 4 factors (COMPLEXITY, ANIMACY, CONCRETENESS, and LENGTH) among the 8 factors were statistically significant.

An interesting fact is that the factor LEVEL was not statistically significant. Though there were some differences among the level of proficiency (Figure 3), this factor LEVEL was not statistically significant as its *p*-value indicates (*p*=.565 and *p*=.349).

### 4.3 Analysis with Effect Plots

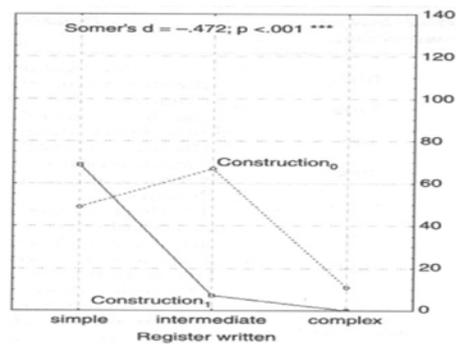
Since the final model was obtained, it is possible to statistically analyze each factor and interactions with effect plots. Among the 8 factors included in the statistical analysis, only 4 main factors were statistically significant. In this section, only those 4 factors were closely examined.

The first factor is COMPLEXITY, which indicates whether the form of DO is simple or complex. Figure 4 is the effect plot for this factor.



**Figure 4.** Effect Plot for COMPLEXITY

Compared this result with that of Gries (2003:194).



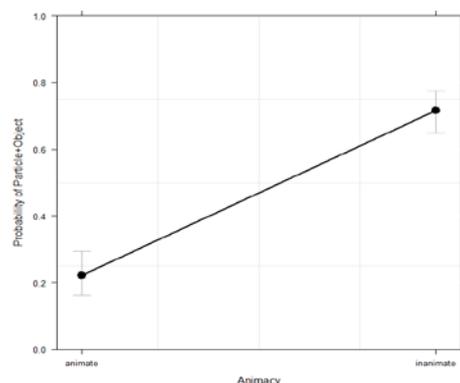
**Figure 5.** COMPLEX in Gries (2003)

COMPLEXITY in this paper corresponds to COMPLEX in Gries (2003).

As Figure 5 demonstrates, the *construction<sub>0</sub>* had higher frequencies than the *construction<sub>1</sub>* when the DO was complex. However, the *construction<sub>1</sub>* had higher frequencies than the *construction<sub>0</sub>* when the DO was simple. This tendency was also observed in Figure 4. When the DO was complex, the proportion of *construction<sub>0</sub>* was greater than the value of *construction<sub>1</sub>*, and its value was greater than 0.5. On the other hand, when the DO was simple, the proportion of *construction<sub>1</sub>* was much greater than the value of *construction<sub>0</sub>*, and its value was less than 0.5. Accordingly, these two graphs demonstrate that native speakers and Korean EFL learners show a similar tendency.

Since two graphs (Figure 4 and Figure 5) are different, it may be unreasonable to compare the values of two graphs. However, since the goal of comparison is to check whether the tendencies that the Korean EFL learners exhibit (not the exact values) are similar to those of native speakers, it is possible to use the analysis results in Gries (2003) in the comparison.

The second factor to be mentioned is ANIMACY, which indicates whether DO was an animate or an inanimate entity. Figure 6 is the effect plot for this factor.



**Figure 6.** Effect Plot for ANIMACY

The y values in this graph represent the ratio of 'Particle + DO'. That is, the y values in this plot represent the ratio of *construction<sub>0</sub>*. Accordingly, as the y value increases, the ratio of *construction<sub>0</sub>* increases. It means that the Korean EFL learners preferred to use *construction<sub>0</sub>*, rather than the ratio of *construction<sub>1</sub>*. It also implies that as the y value increases, the ratio of *construction<sub>1</sub>* decreases.

Now, let's compare this result with that of Gries (2003:197).

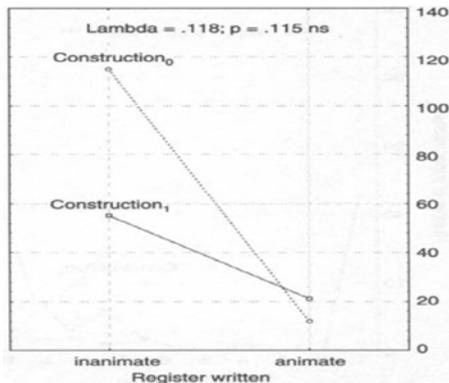


Figure 7. ANIMACY in Gries (2003)

ANIMACY in this paper corresponds to ANIMACY in Gries (2003). The y values in this graph refer to the frequencies of each construction (*construction<sub>0</sub>* and *construction<sub>1</sub>*) when DO has the corresponding value for the given factor.

As Figure 7 demonstrates, though inanimate DOs were prevailed in both constructions, the *construction<sub>1</sub>* has higher frequencies than the *construction<sub>0</sub>* when the DO referred to an animate entity. However, the *construction<sub>0</sub>* has higher frequencies than the *construction<sub>1</sub>* when the DO had an inanimate entity. This tendency was also observed in Figure 6. When DO was inanimate, the proportion of 'Particle + DO' (*construction<sub>0</sub>*) was greater than the value of 'DO + Particle' (*construction<sub>1</sub>*), and its value was greater than 0.5. On the other hand, when DO was an animate entity, the proportion of 'DO + Particle' (*construction<sub>1</sub>*) was greater than the value of 'Particle + DO' (*construction<sub>0</sub>*), and its value was less than 0.5. Accordingly, these two graphs demonstrate the tendency that both native speakers and Korean EFL learners preferred to use *construction<sub>0</sub>* as DO took an inanimate entity.

The third factor is CONCRETENESS, which indicates whether DO refers to an abstract entity or a concrete entity. Figure 8 is the effect plot for this factor.

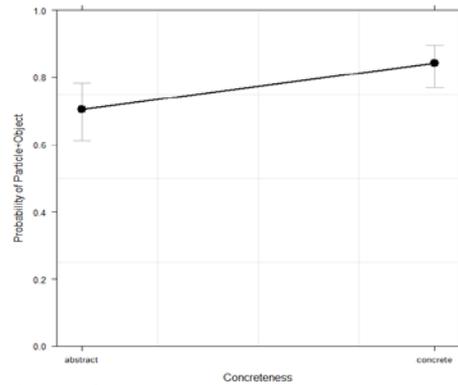


Figure 8. Effect Plot for CONCRETENESS Compared this result with that of Gries (2003:197).

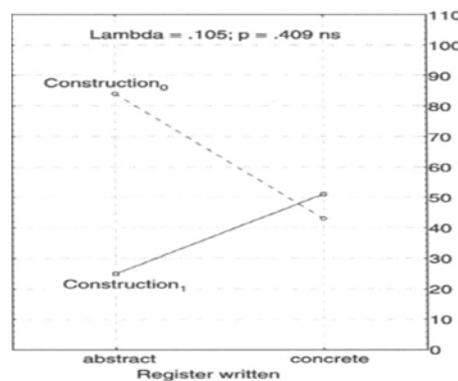


Figure 9. CONCRETE in Gries (2003)

CONCRETENESS in this paper corresponds to CONCRETE in Gries (2003).

As Figure 9 demonstrates, the *construction<sub>1</sub>* has higher frequencies than the *construction<sub>0</sub>* when the DO had a concrete entity. However, the *construction<sub>0</sub>* has higher frequencies than the *construction<sub>1</sub>* when the DO had an abstract entity. This tendency was also observed in Figure 8. When DO had an abstract entity, the proportion of *construction<sub>0</sub>* was greater than the value of *construction<sub>1</sub>*, and its value was greater than 0.5. On the other hand, when DO was a concrete entity, the proportion of *construction<sub>1</sub>* was much greater than the value of *construction<sub>0</sub>*, and its value was less than 0.5. Accordingly, these two graphs demonstrate the tendency that native speakers and Korean EFL learners demonstrate an identical tendency.

The last factor to be mentioned is LENGTH, the length of DO in words. Figure 10 is the effect plot for this factor. Compared this result with that of Gries (2003:194). LENGTH in this paper corresponds to LENTHW in Gries (2003).

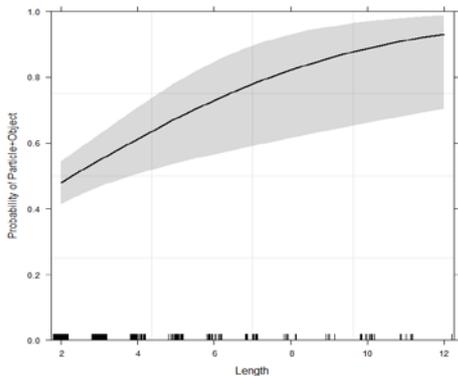


Figure 10. Effect Plot for LENGTH

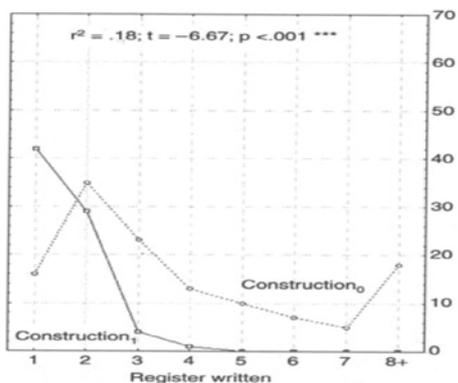


Figure 11. LENGTHW in Gries (2003)

As Figure 11 demonstrates, as DO becomes longer, the *construction<sub>0</sub>* has higher frequencies, while the *construction<sub>1</sub>* has lower frequencies. This tendency was also observed in Figure 10. As DO becomes longer, the ratio of the order *construction<sub>0</sub>* increases. This fact implies that the other order *construction<sub>1</sub>* decreases. Though two graphs were slightly different, both graphs demonstrates the tendency that both native speakers and Korean EFL learners preferred to use *construction<sub>0</sub>* as DO became longer.

## 5 Discussions

In Table 1, several factors were proposed which influenced the alternations of Particle Movement. Among them, 7 factors were chosen for the study in this paper: COMPLEX, ANIMACY, DET, TYPE, IDIOM, CONCRETE, and LENTHW. These factors were encoded as follows: COMPLEXITY, ANIMACY, DEFINITENESS, PRONOMINALITY, IDIOMACITY, CONCRETENESS, and LENGTH. To these 7 factors, one more factor LEVEL was added.

The comparison of Figure 1 and Table 3 demonstrated that the uses of alternation of Particle Movement in Korean EFL learners were different from those of native speakers. Among the 8 factors which influenced alternation of

Particle Movement in native speakers, only 4 factors were statistically significant in Korean EFL learners writings.

Therefore, the answer to the question (2a) will be 'No', and the answer to the question (2b) will be COMPLEXITY, ANIMACY, CONCRETENESS, and LENGTH (4 factors). As for the question (2c), there were some differences in the ratio of these two constructions as the level of proficiency goes up. However, the differences were not statistically significant.

Since Figure 1 and Table 3 demonstrated that Korean EFL learners showed different tendency in the Particle Movement in their writings compared with native speakers, the hypothesis in (3a) cannot be maintained. Instead, the hypothesis in (3b) can be supported, since a different set of factors had influenced Particle Movement in the Korean EFL learners' writings.

## 6 Conclusion

This paper adopted a multifactorial analysis as in Gries (2001, 2003) to examine Particle Movement in Korean EFL learners' writings. The Korean part of TOEFL11 corpus was used, and all the relevant sentences were extracted using the tag information. The eight relevant factors were encoded to these sentences, and each factor and their interactions were statistically analyzed with R.

Through the analysis, it was demonstrated that Korean EFL learners employed a different strategy in the Particle Movement and that only some factors were used for the selection of constructions. Unlike native speakers, 4 linguistic factors were statistically significant in Korean EFL learners' writing samples (ANIMACY, PRONOMINALITY, CONCRETENESS, and LENGTH). It was also observed that there were some differences in the ratio of these two constructions (*construction<sub>0</sub>* vs. *construction<sub>1</sub>*) as the level of proficiency went up. However, the differences were not statistically significant.

However, we do NOT say that these differences between the native speakers and the Korean EFL learners come from only the L1 transfer effects. Another kind of complicated statistical analysis (such as another regression analysis with the native data and/or the analysis in Gries and Deshors (2015)) is necessary to examine if the L1 (here, Korean) really influenced these factors and how much the L1 transfer effects are involved in these factors.

## References

- Daniel Blanchard, Joel Tetreault, Derrick Higgins, Aoife Cahill, and Martin Chodorow. 2013. TOEFL11: A corpus of non-native English. ETS RR-13-24. Princeton, NJ: Educational Testing Service.
- Ping Chen. 1986. Discourse and Particle Movement in English. *Studies in Language* 10:79-95.
- Marcel Den Dikken. 1995. *Particles: On the Syntax of Verb-Particle, Triadic, and Causative Constructions*. Oxford: Oxford University Press.
- Marcel Den Kikken. 1992. *Particles*. Holland Institute of Linguistics Dissertations. The Hague: Holland Academic Graphics.
- Bruce Fraser. 1974. The Phrasal Verb in English, by Dwight Bolinger. *Language*, 50:568-575.
- Bruce Fraser. 1976. *The Verb-Particle Combination in English*. New York: Academic Press.
- Sylviane Granger, Estelle Dagneaux, and Fanny Meynier. 2009. *The international corpus of learner English: Handbook and CD-ROM (version 2)*. Louvain-la-Neuve, Belgium: Presses Universitaires de Lowvain.
- Stephan Th. Gries. 1999. Particle movement: A Cognitive and Functional Approach, *Cognitive Linguistics*, 10(2):105-145.
- Stephan Th. Gries. 2001. A Multifactorial Analysis of Syntactic Variation: Particle Movement Revisited. *Journal of Quantitative Linguistics*, 8(1):33-50.
- Stephan Th. Gries. 2003. *Multifactorial Analysis in Corpus Linguistics: A Study of Particle Movement*. London: Continuum.
- Stephan Th. Gries. 2013. *Statistics for Linguistics with R: A Practical Introduction*. Berlin: Guyter.
- Stephan Th. Gries and Sandra Deshors. 2015. EFL and/vs. ESL? A Multi-level Regression Modeling Perspective on Bridging the Paradigm Gap. *International Journal of Learner Corpus Research* 1(1): 130-159.
- John Hawkins. 1994. *A Performance Theory of Order and Constituency*. Cambridge: Cambridge University Press.
- Otto Jespersen. 1928. *A Modern English Grammar on Historical Principles*. London: George Allen and Unwin Ltd.
- Jurafsky, Daniel and James Martin. 2009. *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics and Speech Recognition*. Upper Saddle Hill, NJ: Prentice Hall.
- Etsko Krusinga and Patrick Erades. 1953. *An English Grammar*. Vol. I. Groningen: P. Noordhoff.
- Yong-hun Lee. 2007. *Corpus Analysis Using NLPTools and Their Applications: Applications to Linguistic Research, English Education, and Textbook Evaluation*. Seoul: Cambridge University Press.
- Bernhard Rohrbacher. 1994. English Main Verbs Move Never. *The Penn Review of Linguistics*, 18:145-159.
- Henry Sweet. 1892. *A New English Grammar*. Oxford: Clarendon Press.
- Rosemary Yeagle. 1983. *The Syntax and Semantics of English Verb-Particle Constructions with off: A Space Grammar Analysis*. Unpublished M.A. Thesis, Southern Illinois University at Carbondale.

# An Efficient Annotation for Phrasal Verbs using Dependency Information

Masayuki Komai

Hiroyuki Shindo

Yuji Matsumoto

Graduate School of Information and Science

Nara Institute of Science and Technology

8916-5, Takayama, Ikoma, Nara, 630-0192, Japan

{komai.masayuki.jy4, shindo, matsu}@is.naist.jp

## Abstract

In this paper, we present an efficient semi-automatic method for annotating English phrasal verbs on the OntoNotes corpus. Our method first constructs a phrasal verb dictionary based on Wiktionary, then annotates each candidate example on the corpus as an either a phrasal verb usage or a literal one. For efficient annotation, we use the dependency structure of a sentence to filter out highly plausible positive and negative cases, resulting in a drastic reduction of annotation cost. We also show that a naive binary classification achieves better MWE identification performance than rule-based and sequence-labeling methods.

## 1 Introduction

Multiword Expressions (MWEs) are roughly defined as those that have “idiosyncratic interpretations that cross word boundaries (or spaces)” (Sag et al., 2001). Vocabulary sizes of single words and MWEs have roughly the same size, thus MWE identification is a crucial issue for deep analysis of natural language text. Indeed, it has been shown in the literature that MWE identification helps various NLP applications, such as information retrieval, machine translation, and syntactic parsing (Newman et al., 2012; Ghoneim and Diab, 2013; Nivre and Nilsson, 2004). Since huge cost is necessary for annotation, there are few corpora that are sufficiently annotated for English MWEs. Schneider et al. (2014b) constructed an MWE-annotated corpus on English Web Treebank, and proposed a sequen-

- (a) We **bring** our computers **up**.
- (b) She **goes over** the question.
- (c) Someone goes over there.

Figure 1: a positive instance (a) of a separable expression “bring up”, a positive instance (b) and a negative instance (c) of an inseparable expression “go over”.

tial labeling method for MWE identification. However, they tried to manually cover the types of comprehensive MWEs, and the number of instances for each MWE was very limited.

In this paper, we propose an efficient annotation method for separable MWEs appearing in a syntactic annotated corpus like the OntoNotes corpus. Although most of natural languages generally have a separable MWEs, an effort for separable MWE annotation is extremely limited. Therefore, we believe that constructing a large-scale corpus for separable MWEs is useful to develop and compare techniques of MWE identification. We especially focus on phrasal verbs that are a majority of separable MWEs, and propose an efficient method for phrasal verb annotation. To efficiently identify MWE usages, we exploit dependency structures on OntoNotes<sup>1</sup>. We also report experiments on MWE identification based on a binary classification, and show that it achieves better performance than rule-based and sequence-labeling methods. Further, we explore effective features for achieving high performance on the MWE identification task.

Our contributions are summarized as follows: (1)

<sup>1</sup>We use English OntoNotes corpus converted into the Stanford Dependency annotation format.

Table 1: The number of MWE types.

	VB	RB	IN	JJ	PRP	DT	<i>Other</i>
types	994	395	94	17	14	12	6
<i>example</i>	go over	far from	in front of	ad hoc	anything else	a few	no way

We propose an efficient semi-automatic method for annotating phrasal verbs on OntoNotes. (2) We show that SVM-based naive classification is sufficient for accurate MWE identification. We also investigate effective features for MWE identification.

## 2 Related Work

MWEs can be roughly divided into two categories, separable and non-separable (or fixed) MWEs. Previous work annotated fixed MWEs on Penn Treebank, where they used syntactic trees of Penn Treebank and an MWE dictionary that is extracted from Wiktionary (Shigeto et al., 2013). In Schneider et al. (2014b), they annotated all types of MWEs on English Web Treebank completely by hand. Afterward, they added to supersenses, which mean coarse-grained semantic classes of lexical units (Schneider and Smith, 2015).

In MWE identification tasks, previous work integrated MWE recognition into POS tagging (Constant and Sigogne, 2011). An MWE identification method using Conditional Random Fields was also presented together with the data set (Shigeto et al., 2013). A joint model of MWE identification and constituency parsing was proposed (Constant et al., 2012). They allocated IOB<sup>2</sup> tags to MWEs and used MWEs as special features when reranking the parse tree. However, it is difficult for these methods to detect discontinuous MWEs.

In contrast, as for methods that can handle separable MWEs, Boukobza and Rappoport (2009) tackled MWE detection on specific MWE types with a binary classification method. In a framework of a sequential labeling method for MWE detection, a new IOB tag scheme, which is augmented to capture discontinuous MWEs and distinguish strong MWEs from weak MWEs, was presented (Schneider et al., 2014a). Here strong MWEs indicate the expres-

sion which has strong idiomaticity, and weak one indicate the expression which is to more likely to be a compositional phrase or collocation. Additionally, words between components of MWEs are called gaps, and the sequential labeling method that allocates IOB tags even to discontinuous sequences. This model is capable of capturing unknown MWEs, but it is difficult to detect new expressions with high accuracy.

## 3 Corpus Annotation

In this section, we present our annotation scheme for phrasal verbs. Our scheme mainly consists of three steps: acquisition of phrasal verbs, identification of phrasal verb occurrences on OntoNotes, and semi-automatic MWE classification with our heuristic rules.

### 3.1 Acquisition of Phrasal Verbs

First, we extract phrasal verb candidates from the English part of Wiktionary<sup>3</sup>. In particular, we parse a dump data of Wiktionary and extract verb entries that are composed of two or more words. We also collect phrasal verb candidates from the Web. For each MWE candidate, we manually check if they actually function as a phrasal verb. Moreover, we manually annotate whether their candidates are “separable” or “inseparable” and whether they are “transitive” or “intransitive”. By “separable”, we mean an object noun phrase can intervene between the main verb and a particle (e.g. **look** the tower **up**). Note that separable phrasal verbs do not always have an intervening object and also that inseparable phrasal verbs can be intervened by an adverb (e.g. **consist** largely **of**).

<sup>2</sup>I, O and B indicate Inside, Outside and Begin in a chunk respectively.

<sup>3</sup><https://en.wiktionary.org/>

Table 2: Statistics in phrasal verb annotation

Annotation type	# instances
manual annotations	4022
automatic annotations	18574
Total	22596

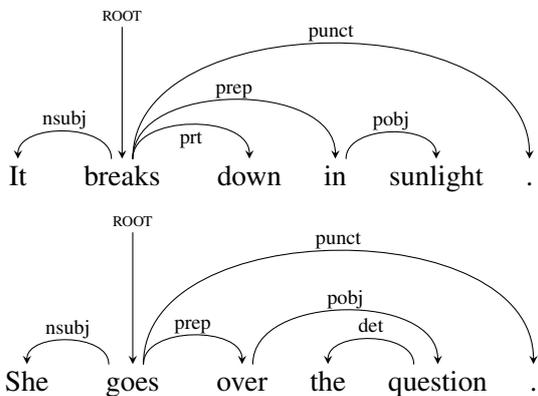


Figure 2: Examples of prt (break down) and prep (go over).

### 3.2 Identification of Phrasal Verb Occurrences in OntoNotes

Second, we retrieve all possible occurrences of phrasal verbs in OntoNotes. Here, we convert a surface word into a lemma form using Python-NLTK<sup>4</sup>, then match the phrasal verb candidates with lemmatized words in OntoNotes. We regard this matching pattern as an instance. In Figure 1, instances (a) and (b) are positive instances where they are used as phrasal verbs. On the other hand, (c) is a negative instance where “go over” is not used as a phrasal verb but is used in the literal meaning. We extract discontinuous patterns as well since phrasal verbs have a potential of appearing discontinuously.

### 3.3 Semi-Automatic Annotation

Third, we check each instance whether it is used as a positive case or a negative one. Since it is too costly to check all instances manually, we propose to make use of dependency structures on OntoNotes to perform semi-automatic annotation. Indeed, we use the Stanford dependency converted from phrase structure trees on OntoNotes corpus. For each possible

<sup>4</sup><http://www.nltk.org/>

Table 3: Annotation rules for phrasal verb candidates. (In this table, “p” is positive, “n” is negative, “m” means manual annotation.)

direct dependency	continuous or not	dependency label	
True	*	prt	p
False	*	prt	n
True	True	prep	p
False	True	prep	m
True	False	prep	m
False	False	prep	n
True	*	other	m
False	*	other	n

instance of a phrasal verb, we use the following relation between the verb and the particle that comprise the phrasal verb candidate: whether the verb and the particle appear adjacently or not and whether the verb and the particle have direct dependency or not, and if so the label of the dependency.

Table 3 shows the whole annotation rules. In these rules, we especially focus on the dependency labels prt and prep in Stanford dependency (de Marneffe and Manning, 2008). The prt label, which directly connects a verb and a particle, may indicate the usage of a phrasal verb, and the prep label indicates a modifier to a verb as a prepositional phrase. Thus, we assume instances which have a direct prt dependency as positive instances. In the case of prep, there is a possibility of phrasal verbs or not. However, we assume instances which are adjacent, have a direct prep relation, and exist in our MWE lexicon as positive instances. If an instance is either not adjacent or having no direct relation with its particle or preposition, we put it for a candidate of manual checking. In this way, we have constructed annotation rules for MWE making full use of syntactic information.

For example, the instance of “break down” in Figure 2 has a direct relation with label prt, thus the first rule in Table 3 is applied. However, there are overlapping ambiguities that are not covered by these rules. For example, an instance “catch up with” can be labeled as positive, but another instance “catch up” of the part of “catch up with”, may also be labeled as positive. When such an ambiguity oc-

Table 4: Corpus statistics of MWEs.

	# instances
positive instances	13214
negative instances	41167
Total	54381

Table 5: Evaluation of annotation rules.

Precision	Recall	F-value
62.72	82.60	71.30

curred, we manually checked their instances. As a result of annotation rules, we could reduce the cost of manual annotation considerably as shown in Table 2.

After annotating phrasal verbs on OntoNotes, we merge our annotation with the fixed MWE annotation done by (Shigeto et al., 2013). However, similar overlapping ambiguities have been generated again in this time (s.t. “get out of” and “out of”), we also manually eliminate these ambiguities.

In Table 4, we show statistics about our constructed corpus after merging. In total, 54381 instances are extracted from the 37015 sentences on OntoNotes,

#### 4 Evaluation of Annotation Rule

In order to evaluate our annotation method, we also validate it on English Web Treebank annotated by (Schneider et al., 2014b). We first apply our method to English Web Treebank, then evaluate the quality of automatic annotation between automatically-annotated positive instances and gold MWEs on English Web Treebank. However, there is a large difference between both MWE candidates since annotators (the dictionary-based rule method and human) and domains of two corpora are different. In view of this, we evaluate only common phrasal verbs between two corpora.

In Table 5 we show the results of evaluating annotation rules. We obtain a sufficient recall, but the precision is lower than we expected. However, we consider this is unavoidable because annotators and domains are different as we have described precisely.

Table 6: The Feature list.  $W, G$  are the position list of the target MWEs to detect and of gaps.  $h$  and  $t$  are the position of the head MWEs and the tail.  $c_i, l_i$  and  $p_i$  is the  $i$ th context word, lemma and POS respectively.  $[c_i]_j^k$  is the substring from  $j$ th to  $k$ th in  $c_i$ .  $F(x)$  is the set that consisted of each element in the  $x$ th feature set.

basic features		
1	$c_i, l_i, p_i$	$ _{i \in W}$
2	$c_i, l_i, p_i$	$ _{i \in W}$
3	$\text{floor}(\frac{ G }{i})$	for $i$ in $\{1, 2, 3, 4, 5\}$
context features		
4	$c_i, l_i, p_i$	$ _{i=h-1}^{h-3}$
5	$c_i, l_i, p_i$	$ _{i=t+1}^{t+3}$
6	$p_i$	$ _{i \in G}$
suffix & prefix features		
7	$[c_i]_1^j$	$ _{j=1}^3$ for $i$ in $h-1$ to $h-3$
8	$[c_i]_j^{ c_i }$	$ _{j= c_i -3}^{ c_i }$ for $i$ in $h-1$ to $h-3$
9	$[c_i]_1^j$	$ _{j=1}^3$ for $i$ in $t+1$ to $t+3$
10	$[c_i]_j^{ c_i }$	$ _{j= c_i -3}^{ c_i }$ for $i$ in $t+1$ to $t+3$
11	$[c_i]_1^j$	$ _{j=1}^3$ for $i$ in $G$
12	$[c_i]_j^{ c_i }$	$ _{j= c_i -3}^{ c_i }$ for $i$ in $G$
combination features		
13	$(e_1, e_2) \in \{F(1) \times F(2)\}$	
14	$(e_1, e_2) \in \{F(1) \times F(3)\}$	
15	$(e_1, e_2) \in \{F(1) \times F(4)\}$	
16	$(e_1, e_2) \in \{F(1) \times F(5)\}$	
17	$(e_1, e_2) \in \{F(1) \times F(6)\}$	
18	$(e_1, e_2) \in \{F(1) \times F(7)\}$	
19	$(e_1, e_2) \in \{F(1) \times F(8)\}$	
20	$(e_1, e_2) \in \{F(1) \times F(9)\}$	
21	$(e_1, e_2) \in \{F(1) \times F(10)\}$	
22	$(e_1, e_2) \in \{F(1) \times F(11)\}$	
23	$(e_1, e_2) \in \{F(1) \times F(12)\}$	

## 5 Experiments

In this section, we evaluate the performance of MWE identification task on our MWE-annotated OntoNotes. The MWE-annotated corpus used in our experiments contains fixed MWE annotations (Shigeto et al., 2013) and our phrasal verb annotations. The corpus is split into 2 sets: 33313 sentences (48970 instances) for training, and 3702 sentences (5411 instances) for testing. In these experiments, the system identifies MWEs given a sentence

Table 7: The experimental results.

	Precision	Recall	F-value
Rule-based method	62.93	<b>97.78</b>	76.58
Augmented IOB (Schneider et al., 2014a)	93.37	91.44	92.40
SVM	<b>93.77</b>	94.27	<b>94.02</b>

with gold POS.

## 5.1 Compared Methods

We compare SVM-based binary classification method against rule-based and sequential labeling method (Schneider et al., 2014a). The SVM method simply classifies each candidate instance as positive case or negative one. For the rule-based method, we use the following two simple rules. The first one is “if the target MWE is an instance of an inseparable phrasal verb and there is no gap between the verb and the particle (or preposition), then it is regarded as positive.” The second one is “if the target MWE is an instance of a separable phrasal verb and the gap is 0 or equal to 1, then it is regarded as positive.” Since our dictionary has information whether the target MWE is separable or not, we can use this information.

Table 6 shows the features that are used for SVM, which are categorized as four types: basic features, context features, suffix & prefix features, and combination features. In this table, bold **c**, **l** and **p** are sequences that are concatenated context words, lemmas, POS sequences of target MWEs respectively. In respect to a classifier, we used SVM<sub>light</sub><sup>5</sup> with a linear kernel.

For sequential labeling method, we follow the previous work (Schneider et al., 2014a) for MWE identification. Their work exploits six types of tags, that is, {**O o B b I i**}, to handle with separable MWE identification, where **O**, **B**, **I** tags indicate **Outside**, **Begin**, **Inside**, and **o**, **b**, **i** tags indicate **outside**, **begin**, **inside** in gaps respectively. In the experiments, we use their implementation<sup>6</sup> with exact match evaluation and set the recall-oriented hyperparameter  $\rho$  to 0.

<sup>5</sup><http://svmlight.joachims.org/>

<sup>6</sup><http://www.cs.cmu.edu/~ark/LexSem/>

Table 8: Investigation of effectiveness of features.

	F-value
<b>basic features</b>	92.89
+ <b>context features</b>	93.69
+ <b>suffix &amp; prefix features</b>	93.06
+ <b>combination features</b>	94.02

## 5.2 Experimental Results

Table 7 summarizes the experimental results. In the table, we can see that SVM-based binary classification outperforms the rule-based and sequential labeling method. This result suggests that simple binary classification is sufficient for accurate MWE identification.

We also investigated which features are effective for our MWE identification task. Table 8 summarizes this analysis result. In the table, we can see that adding context features, suffix & prefix features, and those combinatorial ones to basic features successfully boost the identification performance. Further investigation of combinatorial features could be helpful for achieving better results, but we leave this for future work.

In error analysis, we found it is difficult for our method to detect the mutually-overlapping MWEs. For example, there should be the positive instance of “come out of” and the negative instance of “out of” in nature, but our model may say “positive” for both instances. Resolution of such conflicting cases should be investigated for future work.

Moreover, we have found that it is hard to recognize fixed MWEs, which appear continuously but are in literal usages. For example, “*a bit*” in “*is really a bit player on the stage*” is in literal usages. Our model tends to predict “positive” for such an instance.

## 6 Conclusion

We presented a semi-automatic method for annotating English phrasal verbs on the OntoNotes corpus. For efficient annotation, we use the dependency structure of a sentence to filter out positive and negative cases, resulting in a drastic reduction of annotation cost. We also reported that binary classification method outperformed rule-based and sequential labeling method. In order to improve the accuracy, we need a better model that takes wider contexts into consideration. We consider integration of syntactic parsing into MWE identification is one of such directions.

This paper also have described MWE annotation on OntoNotes. We will make the constructed dataset available on our website<sup>7</sup>. We are hoping that studies on MWEs are increased by using our dataset.

There are MWE types that we haven't handled at this work. For example, some flexible MWEs such as "take into account" are not annotated. Thus, we plan to annotate other discontinuous MWE types on OntoNotes so as to cover all MWEs on OntoNotes. We also believe that MWEs can include syntactic patterns, such as "not only ... but also". To deeply analyze a natural language text, we should explore such directions in future.

## Acknowledgments

We thank the annotator Kayo Yamashita and the anonymous reviewers for their valuable comments. This work has been supported by JSPS KAKENHI Grant Numbers 15K16053 and 26240035. A part of this research was executed under the Commissioned Research of National Institute of Information and Communications Technology (NICT), Japan.

## References

Ram Boukobza and Ari Rappoport. 2009. Multi-word expression identification using sentence surface features. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 468–477.

Matthieu Constant and Anthony Sigogne. 2011. MWU-aware part-of-speech tagging with a CRF model and lexical resources. In *Proceedings of the Workshop on*

*Multiword Expressions: From Parsing and Generation to the Real World*, MWE '11, pages 49–56.

- Matthieu Constant, Anthony Sigogne, and Patrick Watrin. 2012. Discriminative strategies to integrate multiword expression recognition and parsing. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 204–212.
- Marie-Catherine de Marneffe and Christopher D. Manning. 2008. The stanford typed dependencies representation. In *Coling 2008: Proceedings of the Workshop on Cross-Framework and Cross-Domain Parser Evaluation*, CrossParser '08, pages 1–8.
- Mahmoud Ghoneim and Mona Diab. 2013. Multiword expressions in the context of statistical machine translation. In *Proceedings of the Sixth International Joint Conference on Natural Language Processing (IJCNLP)*, pages 1181–1187.
- David Newman, Nagendra Koilada, Jey Lau, and Tim Baldwin. 2012. Bayesian text segmentation for index term identification and keyphrase extraction. In *International Conference on Computational Linguistics (COLING)*, pages 2077–2092.
- Joakim Nivre and Jens Nilsson. 2004. Multiword units in syntactic parsing. In *Workshop on Methodologies and Evaluation of Multiword Units in Real-World Applications*, pages 39–46.
- Ivan A. Sag, Timothy Baldwin, Francis Bond, Ann Copestake, and Dan Flickinger. 2001. Multiword expressions: A pain in the neck for NLP. In *Proc. of the 3rd International Conference on Intelligent Text Processing and Computational Linguistics (CICLING-2002)*, pages 1–15.
- Nathan Schneider and Noah A. Smith. 2015. A corpus and model integrating multiword expressions and supersenses. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL): Human Language Technologies*, pages 1537–1547.
- Nathan Schneider, Emily Danchik, Chris Dyer, and A. Noah Smith. 2014a. Discriminative lexical semantic segmentation with gaps: Running the MWE gamut. *Transactions of the Association of Computational Linguistics (TACL) – Volume 2, Issue 1*, pages 193–206.
- Nathan Schneider, Spencer Onuffer, Nora Kazour, Emily Danchik, Michael T. Mordowanec, Henrietta Conrad, and Noah A. Smith. 2014b. Comprehensive annotation of multiword expressions in a social web corpus. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 455–461.
- Yutaro Shigeto, Ai Azuma, Sorami Hisamoto, Shuhei Kondo, Tomoya Kouse, Keisuke Sakaguchi, Akifumi Yoshimoto, Frances Yung, and Yuji Matsumoto. 2013.

<sup>7</sup><http://cl.naist.jp/en/index.php?Code and Data>

Construction of English MWE dictionary and its application to POS tagging. In *Proceedings of the 9th Workshop on Multiword Expressions*, pages 139–144.

# Color Aesthetics and Social Networks in Complete Tang Poems: Explorations and Discoveries

Chao-Lin Liu<sup>†</sup> Hongsu Wang<sup>‡</sup> Wen-Huei Cheng<sup>§</sup> Chu-Ting Hsu<sup>§</sup> Wei-Yun Chiu<sup>†</sup>

<sup>†§</sup>Department of Computer Science, National Chengchi University, Taiwan

<sup>‡</sup>Institute for Quantitative Social Science, Harvard University, USA

<sup>§</sup><sup>†</sup>Department of Chinese Literature, National Chengchi University, Taiwan

<sup>†</sup>Graduate Institute of Linguistics, National Chengchi University, Taiwan

{<sup>†</sup>chaolin,<sup>§</sup>104753021,<sup>§</sup>whcheng}@nccu.edu.tw, <sup>‡</sup>hongsu.wang@fas.harvard.edu, <sup>†</sup>acwu0523@gmail.com

## Abstract<sup>1</sup>

The *Complete Tang Poems* (CTP) is the most important source to study Tang poems. We look into CTP with computational tools from specific linguistic perspectives, including distributional semantics and collocational analysis. From such quantitative viewpoints, we compare the usage of “wind” and “moon” in the poems of Li Bai<sup>2</sup> (李白) and Du Fu (杜甫). Colors in poems function like sounds in movies, and play a crucial role in the imageries of poems. Thus, words for colors are studied, and “白” (bai2, white) is the main focus because it is the most frequent color in CTP. We also explore some cases of using colored words in antithesis(對仗)<sup>3</sup> pairs that were central for fostering the imageries of the poems. CTP also contains useful historical information, and we extract person names in CTP to study the social networks of the Tang poets. Such information can then be integrated with the China Biographical Database of Harvard University.

## 1 Introduction

*Complete Tang Poems* (CTP) is the single most important collection for studying Tang poems from the literary and linguistic perspectives (Fang et al., 2009; Lee and Wong, 2012). CTP

was officially compiled during the Kangxi years of the Qing dynasty, and includes more than 40,000 poems, totaling more than 3 million characters, that were authored by more than 2000 poets. Employing linguistic theories and computational tools, we analyze the contents of CTP for a wide variety of explorations.

Lo and her colleagues pioneered to handle texts of Chinese classical poetry with computer software (Lo et al., 1997). Hu and Yu (2001) achieved a better environment and demonstrated its functions with a temporal analysis of selected Chinese unigrams, i.e., 愁(chou2), 苦(ku3), 恨(hen4), 悲(bei1), 哀(ai1), and 憂(you1). Jiang (2003) employed tools for information retrieval to find and study selected poems of Li Bai and Du Fu that mentioned “wind” and “moon”. Huang (2004) analyzed the ontology in Su Shi’s (蘇軾) poems based on 300 Tang Poems, and Chang et al. (2005) continued this line of work. Lo then built a more complete taxonomy for Tang poems (Lo, 2008; Fang et al., 2009).

Lee conduct part-of-speech analysis of CTP (Lee, 2012) and dependency trees (Lee and Kong, 2012). They also explored the roles of a variety of named entities, e.g., seasons, directions, and colors, in CTP (Lee and Wong, 2012), and used their analysis of CTP for teaching computational linguistics (Lee et al., 2013).

CTP can serve as the bases of other innovative applications. Zhao and his colleagues have created a website<sup>4</sup> for suggesting couplets, which was accomplished partially based on their analysis of the CTP (Jiang and Zhou 2008; Zhou et al., 2009). Voigt and Jarafsky (2013) considered CTP when they compared ancient and modern verses of China and Taiwan.

Our work is special in that we analyze the contents of CTP from some linguistic perspectives, including collocational analysis and distributional

<sup>1</sup> A majority of the contents of this paper was also published in Chinese in (Liu et al., 2015).

<sup>2</sup> Romanized Chinese names are in the order of surname and first name, following the request of a reviewer.

<sup>3</sup> “Antithesis” is not a perfect translation of “對仗” (dui4 zhang4). Roughly speaking, “對仗” refers to constrained collocations, and requires two terms to have opposite relationships in pronunciations, but does not demand the terms to be opposite in meanings. In English, “antithesis” carries a rather obvious demand for two referred terms to be opposite in meanings.

<sup>4</sup> <http://couplet.msra.cn/> of Microsoft Research Asia



Table 1. Frequent bigrams in CTP

bigram	freq.								
何處	1669	無人	881	青山	662	流水	550	落日	498
不知	1469	風吹	834	少年	634	回首	544	不如	497
萬里	1455	惆悵	780	相逢	629	可憐	539	歸去	496
千里	1305	故人	778	平生	597	如此	526	日暮	496
今日	1165	秋風	749	年年	593	白髮	520	不能	481
不見	1158	悠悠	740	寂寞	592	主人	517	別離	481
不可	1148	相思	733	黃金	589	今朝	516	何時	478
春風	1128	長安	722	天子	588	月明	515	此時	477
白雲	1108	白日	697	人不	587	從此	509	洛陽	476
不得	947	如何	687	天地	586	日月	508	天下	472
明月	896	十年	678	何事	579	行人	507	芳草	472
人間	890	何人	663	江上	553	將軍	499	歸來	471

(bu4 xiang2) means unknown and is not a name. Bai Juyi is the most popular poet in CTP, and has 2643 works, followed by Du Fu's 1158 works and Li Bai's 896 works. In total, we have 3,055,044 Chinese characters and punctuations in this version of CTP.

The exact number of works in CTP needs further research to make sure. Some adjustments should be expected. We cannot determine the authors of some poems even when we read the “御定全唐詩”. The author of the poem with title “題霍山秦尊師” (ti2 huo4 shan1 qin2 zun1 shi1) may be Du Gaung-Ting (杜光庭) or Zheng Ao (鄭遨) as the piece appeared in volumes 854 and 855, respectively.

It is very easy to compute the most common unigrams in CTP, and they are “不”(bu4), “人”(ren2), “山”(shan1), “無”(wu2), “風”(feng1), “一”(yi1), “日”(ri4), “雲”(yun2), “有”(you3), and “何”(he2).

It is very challenging to precisely segment words in poems without human final inspection. Nevertheless, it is well-known that the patterns of 5-character and 7-character Tang poems usually follow specific traditions (Lo, 2005). The segments in poems were constituted by words of one or two characters.

For 5-character Tang poems, the sentences in poems can be segmented into words of 2, 2, and 1 characters or alternatively 2, 1, 2 characters. For instance “白日依山盡”<sup>13</sup> (bai2 ri4 yi1 shan1 jin4) and “感時花濺淚”<sup>14</sup> (gan3 shi2 hua1 jian4 lei4) used, respectively, 2+2+1 and 2+1+2 patterns. Similarly, the sentences of

<sup>13</sup> source: Wang Zhi-Huan (王之渙): 登鶴雀樓 (deng1 guan4 que4 lou2)

<sup>14</sup> source: Du Fu (杜甫): 春望 (chun1 wang4)

7-character Tang poems usually used 2+2+2+1 and 2+2+1+2 patterns, e.g., “東風不與周郎便”<sup>15</sup> (dong1 feng1 bu4 yu3 zhou1 lang2 bian4) and “晉代衣冠成古丘”<sup>16</sup> (jin4 dai4 yi1 guan1 cheng2 gu3 qiu1), respectively.

Employing such a literary common sense as a heuristic for segmenting words in CTP, we can find words of relatively high frequencies in Table 1. In the order of higher frequencies, the most common two-character words in CTP are “何處” (he1 chu4, where), “不知” (bu4 zhi1, unknown), “萬里” (wan4 li3, tens of thousands of miles), “千里” (qian1 li3, thousands of miles), “今日” (jin1 ri4, today), “不見” (bu2 jian4, cannot be seen), “不可” (bu4 ke3, cannot), “春風” (chun1 feng1, spring wind), “白雲” (bai2 yun2, white cloud), “不得” (bu4 de2, cannot), “明月” (ming2 yue4, bright moon) 和 “人間” (ren2 jian1, human world).

Not all frequent strings thus identified are real words. “人不” (ren2 bu4) in Table 1 is not a word but just a frequent string as in “盡日傷心人不見”<sup>17</sup> (jin4 ri4 shang1 xin1 ren2 bu2 jian4) and “雖病人不知”<sup>18</sup> (sui1 bing4 ren2 bu4 zhi1).

Proposing nonwords like “人不” brings researchers inconvenience but does not cause serious troubles. Considering that we are handling millions of characters in CTP and that it is neither impossible nor very time-consuming to

<sup>15</sup> source: Li Bai (李白): 登金陵鳳凰臺 (deng1 jin1 ling2 feng4 huang2 tai2)

<sup>16</sup> source: Du Mu (杜牧): 赤壁 (chi4 bi4)

<sup>17</sup> source: Li Shang-Yin (李商隱): 遊靈伽寺 (you2 ling2 qie2 si4)

<sup>18</sup> source: Bai Juyi (白居易): 讀史五首 (du2 shi3 wu3 shou3)

Table 2. Li Bai's wind

bigram	freq.								
春風	72	松風	17	南風	8	悲風	6	高風	4
清風	28	隨風	14	北風	8	飄風	5	西風	4
秋風	26	香風	11	涼風	8	胡風	5	扶風	4
東風	24	天風	10	狂風	7	從風	5	屏風	4
長風	22	英風	8	雄風	6	巖風	5	動風	4

Table 3. Du Fu's wind

bigram	freq.								
秋風	30	朔風	8	高風	6	江風	4	南風	4
春風	19	微風	8	清風	6	驚風	4	涼風	4
北風	14	隨風	7	天風	6	山風	4	東風	4
悲風	10	回風	7	長風	5	多風	4		
裡風	8	臨風	7	陰風	4	含風	4		

identify such nonwords, our experience indicates that applying computational tools for automatic processing of the CTP contents made our study of CTP more efficient than without using the tools.

We have designed tools for extracting contexts that contain candidate words for researchers' inspections, so validating candidate words is very easy. Furthermore, sometimes the words that do not exist in contemporary literature may turn out to be artistic usages of words that are hard to be correctly handled by current tools for Chinese segmentation, and researchers are more than happy to check those innovative words with a limited cost in time.

Table 1 exemplifies a problem of this paper: that we cannot provide pronunciations for all Chinese words that appear in this page-limited manuscript. Due to the large number of Chinese words in this and other tables, we cannot afford to annotate and explain all of the words in tables.

### 3 Styles

Adapting the concept of distributional semantics<sup>19</sup> (Harris, 1954; Miller and Walter 1991), researchers investigate the semantics of a word from the distributions of its surrounding words. Firth (1957) stated that **"You shall know a word by the company it keeps."** Similarly, we should be able to extend the concept of distributional semantics to compare poets' styles: **"You shall know a poet's style by the words s/he uses."**

<sup>19</sup> A reviewer for this paper, referring to (Lin, 1998), considers our approach as a collocational analysis of words in a given context. We used "distributional" because there is a sense of distribution when we can consider the frequency distribution of a set of words.

### 3.1 Wind and Moon

Jiang (2003) compared the styles of Li Bai and Du Fu by looking into how they used "風" (feng1, wind) and "月" (yue4, moon) in their works by checking into individual pieces of poems. We take a quantitative approach by examining the terms related to "風" and "月" in the poets' works.

We can employ the PAT-tree techniques (Chien, 1997) or our own tools<sup>20</sup> for finding the characters that appear immediately before a specified Chinese character to find words containing "風" and "月" in poets' works.

Tables 2 and 3 show how Li and Du used "wind." Both Li and Du were very creative in inventing terms for "風". The most common term of "風" in Li Bai's works were "春風" (chun1 feng1, spring wind), "清風" (qing1 feng1, clear wind), "秋風" (qiu1 feng1, autumn wind), "東風" (dong1 feng1, eastern wind), and "長風" (zhang2 feng1, long wind). Du Fu, on the other hand, had "秋風", "春風", "北風" (bei3 feng1, northern wind), "悲風" (bei1 feng1, sad wind), and "朔風" (shuo4 feng1, northern wind).

For those who can appreciate Chinese, the poets' "風" carried very different imageries. For instance, in China, winds from the north are generally cold, which has been used to convey a sense of sadness by many. In contrast, the eastern wind and spring wind are more comfortable and pleasing.

Tables 4 and 5 show how Li and Du used "moon." Although Du had more works in CTP

<sup>20</sup> <https://sites.google.com/site/taiwandigitalhumanities/>

**Table 4. Li Bai’s moon**

bigram	freq.								
明月	57	溪月	9	有月	5	湖月	3	夜月	3
秋月	40	八月	9	轉月	4	漢月	3	夕月	3
五月	28	雲月	9	曉月	4	樓月	3	喘月	3
日月	23	花月	8	孤月	4	新月	3	向月	3
海月	14	見月	7	台月	4	待月	3	古月	3
上月	13	江月	6	落月	3	弄月	3	十月	3
三月	13	蘿月	5	片月	3	如月	3	二月	3
山月	10	素月	5	滿月	3	好月	3	乘月	3

**Table 5. Du Fu’s moon**

bigram	freq.								
日月	20	明月	7	落月	4	正月	3	從月	3
歲月	14	江月	6	秋月	4	星月	3	九月	3
十月	10	五月	6	漢月	4	新月	3		
三月	9	夜月	5	門月	3	四月	3		
八月	8	二月	5	素月	3	六月	3		

(cf. Section 2.2), statistics in Tables 4 and 5 show that Li used more words about “月” than Du. Li and Du also demonstrated quite diverging styles in using “月”. For frequent terms of “月”, Li had “明月” (ming2 yue4, bright moon), “秋月” (qiu1 yue4, autumn moon), “日月” (ri4 yue4, sun and moon), “海月” (hai3 yue4, sea and moon), and “山月”(shan1 yue4, mountain moon); in contrast, Du used more “歲月” (sui4 yue4, ages) and monthly names, e.g., “十月” (shi2 yue4, October), “三月” (san1 yue4, March), and “八月” (ba1 yue4, August).

### 3.2 White Words in CTP

Colors for poems are like sounds for movies. They foster the feelings and imageries of the artistic works. When we checked the most frequent unigrams in CTP (cf. Section 2.2), we have found that “白” (bai2, white) is the most frequent color name in CTP.

Using the same mechanism for finding words that contained “風” and “月” in CTP, we can find words that started with “白”. Then, we can calculate the percentage of a poet’s poems that used specific words that started with “白”. The statistics are collected for 13 renowned poets and are listed in Table 6, which is placed at the end of this manuscript because of its huge size.

The third and following rows of Table 6 show two types of information. The second leftmost column lists the white words that appeared more than 30 times in the works of 12

poets<sup>21</sup>. The leftmost column lists the total frequencies of these white words that were used in the poets’ poems. The percentages that appear to the right of the white words indicate how often an individual poet used this white word, while the thick boxes indicate the most frequent words used by the poets.

The first row, Ratio A, of Table 6 shows the total percentage of the poet’s poems in CTP that used the white words in Table 6. Li Bai liked to use “白” much more than others based on the data. Nearly half of Li’s poems in CTP, i.e., 46.65%, used the color.

The second row, Ratio B, of Table 6 shows the total percentage of poets’ poems that used “白髮” (bai2 fa3, gray hair), “白頭”(bai2 tao2, white head), “白首” (bai2 shou3, white head), “白鬚”(bai2 syu1, white beard), “白骨”(bai2 gu3, white bone), and “白髭” (bai2 zi1, white mustache). Statistics for these terms are specially marked by shadowed rows. These six terms typically appeared in works that carried pessimistic senses.

It is thus possible to peek into the differences of the main themes of poets’ works with Ratio B. The B ratios of Meng Hao-Ran (孟浩然), Li Shang-Yin (李商隱), and Wen Ting-Yun (溫庭筠) are less than 2%. In sharp contrast, we could see that the B ratios of Du Fu and Bai Juyi are more than 7%. In general, Meng’s works are

<sup>21</sup> Table 6 also includes statistics for selected words that appeared more less than 30 times. This table is adapted from a similar table in (Liu et al., 2015).

**Table 6. Percentages of poets' works that used white words**

Ratio A		8.96	18.41	9.73	46.65	23.83	12.55	26.94	15.67	18.80	17.37	16.30	10.48
Ratio B		1.87	5.72	1.80	5.92	2.13	4.66	7.94	1.99	2.28	7.19	3.70	3.23
freq.	bigram	孟浩然	孟郊	李商隱	李白	李賀	杜牧	杜甫	溫庭筠	王維	白居易	賈島	韓愈
217	白日	0.75	4.73	1.62	6.92	2.98	1.01	2.42	0.00	1.14	2.04	2.22	3.23
164	白髮	1.12	3.73	0.54	2.34	1.28	1.62	1.99	0.00	0.85	2.50	2.22	0.54
158	白雲	2.99	1.99	0.54	3.79	0.85	1.42	0.86	0.28	7.41	0.95	4.44	0.27
149	白頭	0.00	0.75	0.72	0.67	0.43	2.23	3.37	1.14	0.57	2.23	0.49	1.61
86	白首	0.75	1.00	0.18	1.56	0.43	0.20	1.99	0.85	0.85	1.02	0.25	0.81
74	白玉	0.00	0.50	2.34	3.01	0.85	0.81	0.60	0.00	0.85	0.53	0.00	0.27
74	白馬	0.37	0.50	0.00	2.34	4.68	0.00	1.38	2.85	0.85	0.30	0.00	0.00
63	白雪	0.37	0.25	0.36	2.34	0.00	0.40	1.04	0.28	0.00	0.68	0.00	0.27
59	白帝	0.00	0.00	0.18	1.00	0.43	0.00	3.54	0.28	0.00	0.08	0.00	0.54
58	白露	0.00	0.50	0.18	1.56	0.43	0.00	0.86	0.28	0.28	0.79	0.99	0.27
54	白石	0.00	1.00	0.90	1.12	0.43	0.00	0.26	0.57	0.57	0.68	1.23	0.81
38	白蘋	0.37	0.75	0.18	0.22	1.28	0.20	0.52	2.85	0.00	0.30	0.00	0.54
32	白水	0.00	0.25	0.00	0.89	1.28	0.00	1.12	0.00	0.57	0.08	0.00	0.27
31	白蘋	0.00	0.00	0.18	0.11	0.00	0.20	0.00	0.00	0.00	0.98	0.49	0.00
30	白鷺	0.00	0.25	0.00	1.79	0.00	0.40	0.26	0.00	0.85	0.15	0.25	0.00
21	白骨	0.00	0.25	0.18	1.23	0.00	0.00	0.60	0.00	0.00	0.00	0.00	0.27
17	白衣	0.00	0.00	0.18	0.22	0.00	0.00	0.17	0.00	0.57	0.19	0.99	0.00
15	白髭	0.00	0.00	0.00	0.00	0.00	0.40	0.00	0.00	0.00	0.45	0.25	0.00
15	皓齒	0.00	0.00	0.00	0.78	0.85	0.00	0.26	0.85	0.00	0.00	0.00	0.00

considered to belong to the relaxing category (田園詩派, Tian2 Yuan2 Shi1 Pai4), and Li and Wen are considered to use expressions that lead to “a beautiful and gorgeous conception” (Lee, 2009). Both Du Fu and Bai Juyi, on the other hand, are considered as social poets who cared about the status of the society.

#### 4 Collocations and Antithesis

Collocations refer to the occurrence of two words within a specified range. If two words constantly appear within a short range, they may have some close relationships in semantics.

Antithesis (對仗, dui4 zhang4) refers to the parallelism in poems. Two words need to have special relationships in their positions, pronunciations, and meanings to form an antithesis pair.

There are complex rules<sup>22</sup> for observing antitheses and rhymes in Tang poems. In a Lu Shi (律詩), the third and the fourth sentences form a sentence pair (聯, lian2), so do the fifth and the sixth sentences. A pair of sentences should follow the antithesis rules. In “白日當空天氣暖, 好風飄樹柳陰涼”<sup>23</sup>, “白日” and “柳陰”

form a collocation but not antithesis, while “白日” and “好風” are both a pair of collocation and a pair of antithesis. In this case, “白日” and “柳陰” do not locate at corresponding positions.

If we can segment words in poems correctly, then recognizing antithesis will not be very difficult. However, perfect word segmentation in poems needs semantic information. Sometimes the poems could carry ambiguous meanings.

##### 4.1 Word Pairs

We can compute the collocations of a word to acquire a sense of the circumstances of the word's occurrences. To do so, we extract contexts of words, say *n* characters, before and after the word of interest from CTP. Then we can compute frequent words from the contexts (cf. Section 2.2).

Table 7 lists some educational findings when we set *n* to 30. It is very interesting to find out that “白雲” (bai2 yun2, white cloud) collocates with “明月” (ming2 yue4, bright moon) and “流水” (liu2 shui3, running water), that “白日” (bai2 ri4, bright sun) collocates with “青春”

<sup>22</sup> <http://cls.hs.yzu.edu.tw/300/all/primary1/DET4.htm>

<sup>23</sup> source: Yuan Zhen (元稹): 清都春霽, 寄胡三、

吳十一 (qing1 dou1 chun1 ji4, ji4 hu2 san1 - wu2 shi2 yi1)

Table 7. Statistics of some collocations in CTP (n=30)

白雲				白日		白髮			
bigram	freq.								
明月	61	清露	10	青春	32	青山	38	丹砂	7
流水	40	青壁	7	青山	21	青雲	27	黃河	6
芳草	29	秋草	7	清風	18	朱顏	16	清光	4
滄海	28	丹灶	5	紅塵	15	青春	15	丹霄	4
紅葉	17	青鏡	2	黃河	15	黃金	13	黃衣	3
黃葉	16	青玉	2	滄江	6	滄洲	8	紅塵	3
青草	14	皇道	1	青蓮	3	青衫	7	紅旗	3
				青霄	3				
				青楓	2				

Table 8. Corresponding colors in CTP

白		青		紅		黃		綠		紫		碧		丹		赤		黑	
C	F	C	F	C	F	C	F	C	F	C	F	C	F	C	F	C	F	C	F
青	919	白	919	白	358	白	505	紅	335	青	197	紅	199	白	142	青	54	青	36
黃	505	綠	202	綠	335	青	152	青	202	黃	139	青	188	紫	70	黃	39	黃	27
紅	358	紫	197	碧	199	紫	139	黃	83	紅	107	清	100	碧	50	白	33	紅	24
清	274	碧	188	翠	139	綠	83	白	70	白	72	黃	74	青	41	紫	19	白	15
丹	142	黃	152	青	111	碧	74	清	70	丹	70	白	57	翠	35	蒼	15	明	13
滄	99	紅	111	紫	107	紅	44	丹	31	清	56	丹	50	綠	31	紅	13	清	10
朱	97	翠	54	黃	44	赤	39	朱	27	朱	41	金	42	玉	29	滄	12	丹	8
明	96	赤	54	清	36	翠	33	紫	26	金	39	紫	35	素	25	丹	10	寒	8
綠	70	明	42	素	31	清	32	碧	26	碧	35	朱	31	金	21	清	8	紫	7
玄	66	丹	41	金	21	黑	27	金	23	玄	32	寒	22	清	17	朱	7	赤	7

(qing1 chun1, young age) and “青山” (qing1 shan1, mountains), and that “白髮” (bai2 fa3, gray hair) collocates with “青山” and “青雲” (qing1 yun2, blueish clouds).

We can also study the cases of antitheses of the white words that were used by individual poets. For instance, we can find at least 26 instances of “白髮” and “青雲” that were used as a antithesis pair in CTP. “白雲” and “流水” were used as a antithesis pair by Liu Yu-Xi (劉禹錫), Yao He (姚合), Huang-fu Ran (皇甫冉), Huang-fu Cent (皇甫曾), Jia Dao (賈島), and Qian Qi (錢起), while “白雲” and “青草” (qing1 cao3, green grass) were used as a antithesis pair by Liu Zhang-Qing (劉長卿), Shu-kong Si (司空曙), Yao Ho (姚合), Zhang Ji (張籍), Li Tuan (李端), and Lang Shi-Yuan (郎士元).

These statistics offer some hints about the word semantics, and researchers may want to (and we can) extract the poems that contain a specific pair of words to examine the complete poems for either literary or social studies.

#### 4.2 More Colors in CTP

It is certainly possible to focus on one-character color words as well. We can check the positions of the colors in sentences, and find pairs of colors that appear at the same corresponding positions in a pair of sentences. As the most common color in CTP, “白” corresponds to many other colors: “朱”(zhu1), “丹”(dan1), “紅”(hong2), “緋”(fei1), “彤”(tong2), “青”(qing1), “翠”(cui4), “碧”(bi4), “綠”(lu4), “蒼”(cang1), “清”(qing1), “紫”(zi3), “玄”(xuan2), “皂”(zao4), “黑”(hei1), “淩”(lu4), “明”(ming2), “黃”(huang2), “金”(jin1), and “銀”(yin2).

Table 8 lists some of the frequent color pairs for 10 colors in 10 major columns, each separated by a double bar. In each major column, the C sub-column lists the colors that correspond to the color of the major column, and the F sub-column shows the frequencies.

That “白” corresponds to “青” (qing1, blue) and “黃” (huang2, yellow) and that “碧” (bi4, green) corresponds to “紅” (hong2, red) and “青” most frequently are s interesting findings.

Given these statistics and other computational supports, we are ready to explore more interesting topics that are related colors in CTP (Cheng et al., 2015).

## 5 Social Network Analysis

Poets mentioned names of their friends or other people in the titles and contents of their poems, so we can use the CTP as a basis for studying social networks of Tang poets. As an extreme example, Li Bai mentioned himself in his own poems: “李白乘舟將欲行，忽聞岸上踏歌聲”<sup>24</sup> (li3 bai2 cheng2 zhou1 jiang1 yu4 xing2, hu1 wen2 an4 shang4 ta4 ge1 sheng1).

In CTP, at least eight poets mentioned Li Bai in 15 works, among which Du Fu contributed seven. We can also see comments on Du Fu by Luo Yin (羅隱), i.e., “杜甫詩中韋曲花，至今無賴尚豪家”<sup>25</sup> (du4 fu3 shi1 zhong1 wei3 qu3 hua1, zhi4 jin1 wu2 lai4 shang4 hao2 jia1).

Of course, mentioning a person’s name may not imply direct friendship. The title “長沙過賈誼宅” (chang1 sha1 guo4 jia3 yi2 zhai2) cannot be used to infer that Liu Zhang-Qing (劉長卿), the author, passed Jia Yi’s (賈誼) home, which is almost impossible as Jia passed away in 168 BC, and Liu was born in 709 AD.

It is easy to build the relationship of “mentioning the name of” in poems, but it takes more discretion to judge direct friendships. We can employ other biographical information such as style names (字, zi4), pen names (號, hao4), birthdays of the poets, from the China Biographical Database<sup>26</sup> to make reliable decisions.

Sometimes, a poet’s name is not completely listed in poems. In “白也詩無敵，飄然思不群”<sup>27</sup> (bai2 ye3 shi1 wu2 di2, piao1 ran2 si1 bu4 qun2), Du Fu referred to Li Bai only by “白”. Hence, cares are needed to handle special cases.

Verbs can offer extra information about the relationships between the poets and the mentioned persons. For instance, Li Shi-Min (李世民) was a Tang emperor, and he “賜” (ci4, give as a present) his poems to officers. Du Fu would “憶” (yi4, recall) Li Bai. Such verbs show us not only the way to find connections between persons but also the types of connections. Poems of

emperors, for instance, shed light on their connections with high-ranking officers that are useful for historical studies (cf. Chen, 2010).

We may request a list of such special verbs from domain experts, or we may apply the technique of “word clippers” (Chang, 2006) to find verbs that collocated with names, thus providing opportunities for finding diverse, realistic and virtual connections among poets.

## 6 Concluding Remarks

Finding a needle in a haystack is challenging for human beings, but finding specific words in millions of words is easy for computers. With the aforementioned applications, we demonstrated the potentials of computational tools for studying the Complete Tang Poems, which is a bright spot in a fast-growing research field – Digital Humanities. Computational tools, such as information retrieval, textual analysis, and text mining, cannot accomplish deep research yet, but they can help researchers find and collect much more relevant research material with astonishing efficiency.

Evidence shows that knowing the collection of words that were used by individual poets opens a window for observing the inner worlds of the poets. The concept of distributional semantics is proved to be effective for studying CTP.

We still need to strengthen our ability to check the constraints for pronunciations and rhymes in poems so that we can judge antithesis more precisely with less human participation.

The functions of colors in poems offer a stimulating direction that we intend to dig in further. To do so, we need to employ more technologies for affective computing (Zheng, 2012) so that our software can learn to read between the lines.

We are grateful to the reviewers of this paper for valuable pointers for collocation networks (Williams, 1998) and style analysis (Quiniou et al., 2012). We will have to consider these suggestions in the context of CTP, which contains limited material for individual poets. The actual work about building social networks with CTP is still underway.

## Acknowledgements

This work was supported in part by the Ministry of Science and Technology of Taiwan under grants MOST-102-2420-H-004-054-MY2 and MOST-104-2221-E-004-005-MY3.

<sup>24</sup> source: Li Bai (李白): 贈汪倫 (zeng4 wang1 lun2)

<sup>25</sup> source: 寄南城韋逸人 (ji4 nan2 cheng2 wei3 yi4 ren2)

<sup>26</sup> <http://isites.harvard.edu/icb/icb.do?keyword=k35201>

<sup>27</sup> source: Du Fu (杜甫): 春日憶李白 (chun1 ri4 yi4 li3 bai2)

## References

- Chang, Shan-Pin (張尚斌). 2006. *A Word-Clip algorithm for Named Entity Recognition - by example of historical documents*, Master's thesis, National Taiwan University, Taiwan. (in Chinese)
- Chang, Ru-Yng, Chu-Ren Huang, Fengju Lo, and Sueming Chang. 2005. From general ontology to specialized ontology: A study based on a single author historical corpus, *Proc. of the Workshop on Ontologies and Lexical Resources*, 16–21.
- Chen, Jack Wei. 2010. *The Poetics of Sovereignty: On Emperor Taizong of the Tang Dynasty*, Harvard University Asia Center, 2010.
- Cheng, Wen-Hui, Chao-Lin Liu, Chu-Ting Hsu, and Wei-Yuan Chiu. 2015. Sentiment phenomenology and color politics: Observations of white words in mid-Tang poems, under review.
- Chien, Lee-Feng. 1997. PAT-tree-based keyword extraction for Chinese information retrieval, *Proc. of the 20th Annual Int'l ACM SIGIR Conf. on Research and Development in Information Retrieval*, 50–58.
- Fang, Alex Chengyu, Fengju Lo, and Cheuk Kit Chinn. 2009. Adapting NLP and corpus analysis techniques to structured imagery analysis in classical Chinese poetry, *Proc. of the Workshop on Adaptation of Language Resource and Technology to New Domains*, 27–34.
- Firth, John Rupert. 1957. *A synopsis of linguistic theory 1930–1955*, *Studies in Linguistic Analysis*, 1–32.
- Harris, Zellig. 1954. Distributional structure, *Word*, 10(2-3):1456–1162.
- Hu, Junfeng (胡俊峰) and Shiwen Yu (俞士汶). 2001. The computer aided research work of Chinese ancient poems, *ACTA Scientiarum Naturalium Universitatis Pekinensis*, 37(5):725–733. (in Chinese)
- Huang, Chu-Ren. 2004. Text-based construction and comparison of domain ontology: A study based on classical poetry, *Proc. of the 18th Pacific Asia Conf. on Language, Information and Computation*, 17–20.
- Jiang, Long and Ming Zhou. 2008. Generating Chinese couplets using a statistical MT approach, *Proc. of the 22nd Int'l Conf. on Computational Linguistics*, 377–384.
- Jiang, Shao-Yu (蔣紹愚). 2003. “Moon” and “Wind” in Bai Li's and Fu Du's poems – Using computers for studying classical poems, *Proc. of the 1st Int'l Conf. on Literature and Information Technologies*. (in Chinese)
- Lee, John. 2012. A classical Chinese corpus with nested part-of-speech tags, *Proc. of the 6th EACL Workshop on Language Technology for Cultural Heritage, Social Sciences, and Humanities*, 75–84.
- Lee, John, Ying Cheuk Hui, and Yin Hei Kong. 2013. Treebanking for data-driven research in the classroom, *Proc. of the 4th Workshop on Teaching Natural Language Processing*, 56–60.
- Lee, John and Yin Hei Kong. 2012. A dependency treebank of classical Chinese poems, *Proc. of the 2012 Conf. of the North Chapter of the Association for Computational Linguistics: Human Language Technologies*, 191–199.
- Lee, John and Tak-sum Wong. 2012. Glimpses of ancient China from classical Chinese poems, *Proc. of the 24th Int'l Conf. on Computational Linguistics*, posters, 621–632.
- Lee, Wei-Chih (李瑋質). 2009. *Wen Ting-Yun and Li Shan-Yin's works in the late Tang receive to the Gong-Ti Poetry of the Southern Dynasties*, Master's thesis, National Central University, Taiwan. (in Chinese)
- Lin, Dekang. 1998. Automatic retrieval and clustering of similar words. *Proc. of the 36th Annual Meeting of the Association for Computational Linguistics*, 768–774.
- Liu, Chao-Lin, Chun-Ning Chang, Chu-Ting Hsu, Wen-Huei Cheng, Hongsu Wang, and Wei-Yuan Chiu. 2015. Textual Analysis of Complete Tang Poems for Discoveries and Applications — Style, Antitheses, Social Networks, and Couplets, to appear in *Proc. of the 27th Conf. on Computational Linguistics and Speech Analysis*. (in Chinese)
- Lo, Fengju (羅鳳珠). 2005. Design and applications of systems for word segmentation and sense classification for Chinese poems, *Proc. of the 4th Conference on Technologies for Digital Archives*. (in Chinese)
- Lo, Fengju. 2008. The research of building a semantic category system based on the language characteristic of Chinese poetry, *Proc. of the 9th Cross-Strait Symposium on Library Information Science*. (in Chinese)
- Lo, Fengju, Yuanping Li (李元萍), and Weizheng Cao (曹偉政). 1997. A realization of computer aided support environment for studying classical Chinese poetry, *J. of Chinese Information Processing*, 1: 27–36. (in Chinese)
- Miller, George and Walter Charles. 1991. Contextual correlates of semantic similarity, *Language and Cognitive Processes*, 6:1–28.
- Quiniou, Solen, Peggy Cellier, Thierry Charnois, Dominique Legallois. 2012. What about sequential data mining techniques to identify linguistic patterns for stylistics? *Proc. of the 13th Int'l Conf. on Intelligent Text Processing and Computational Linguistics*, 166–177.

- Voigt, Rob and Dan Jurafsky. 2013. Tradition and modernity in 20th century Chinese poetry, *Proc. of the 2nd Workshop on Computational Linguistics for Literature*, 17–22.
- Williams, Geoffrey. 1998. Collocational networks: Interlocking patterns of lexis in a corpus of plant biology research articles
- Zheng, Yongxiao (鄭永曉). 2012. Affective computing applied in Chinese classical poetry, *E-Science: Technology & Application*, 3(4):59–66. (in Chinese)
- Zhou, Ming, Long Jiang, and Jing He. 2009. Generating Chinese couplets and quatrain using a statistical approach, *Proc. of the 23rd Pacific Asia Conf. on Language, Information and Computation*, 43–52

# Korean Twitter Emotion Classification Using Automatically Built Emotion Lexicons and Fine-Grained Features

**Hyo Jin Do and Ho-Jin Choi**

School of Computing  
Korea Advanced Institute of Science and Technology  
Daejeon, Republic of Korea  
{hjdo, hojinc}@kaist.ac.kr

## Abstract

In recent years many people have begun to express their thoughts and opinions on Twitter. Naturally, Twitter has become an effective source to investigate people's emotions for numerous applications. Classifying only positive and negative tweets has been exploited in depth, whereas analyzing finer emotions is still a difficult task. More elaborate emotion lexicons should be developed to deal with this problem, but existing lexicon sets are mostly in English. Moreover, building such lexicons is known to be extremely labor-intensive or resource-intensive. Finer-grained features need to be taken into account when determining finer-emotions, but many existing works still utilize coarse features that have been widely used in analyzing only the polarity of emotion. In this paper, we present a method to automatically build fine-grained emotion lexicon sets and suggest features that improve the performance of machine learning based emotion classification in Korean Twitter texts.

## 1 Introduction

Nowadays, people freely express their thoughts on microblogs, and Twitter is known to be one of the popularly used services. In 2014, 500 million tweets were sent per day by 316 million monthly active users across the globe<sup>1</sup>. Not surprisingly, Twitter has been actively mined in the field of computer science to investigate public opinion (Diakopoulos and Shamma, 2010; Kim et al., 2014; O'Connor et al.,

2010), get real-time information (Doan et al., 2012), and even forecast future events (Bollen et al., 2011). All such research shows Twitter's potentials in the analysis of human thought and behavior. In particular, researchers are showing interest in the analysis of human emotions presented in Twitter messages. Many studies have been done to classify sentiments (positive and negative) in tweets. Going further, researchers are currently trying to analyze fine-grained emotions beyond polarity. Fine-grained emotion analysis is known to be more challenging than sentiment analysis because it must identify subtle differences between emotions. Dealing with emotions in an individual Twitter post is even more difficult because of its short length with the frequent use of informal words and erroneous sentence structures. Elaborate emotion lexicons should be used to deal with the problem, but non-English speaking countries have difficulties using existing lexicon sets because they are mostly in English. Further, building such lexicons is known to be extremely labor-intensive or resource-intensive that can be a burden to under-resourced countries. Moreover, a set of features that achieves the best performance in fine-grained emotion classification should be exploited that is particularly attuned to tweets written in specific language.

Our goal in this paper is to classify Korean Twitter messages into fine-grained emotions. The emotion types are Ekman's six basic emotions (Ekman, 1992) and it is known to be the most frequently used in the field of computer science for emotion mining and classification (Bann and Bryson, 2012). For this goal, we employed machine learning algorithms

<sup>1</sup><https://about.twitter.com/company>

with fine-grained features including an emotion lexicon feature. Specifically, we addressed the following problems:

1. *Emotion lexicon construction.* Is there any simple and automatic method to generate emotion lexicons particularly attuned to the Twitter domain without using other lexical resources?
2. *Feature engineering.* What is the best set of features that can effectively show the subtle distinctions between finer-grained emotions expressed in Korean Twitter texts?

We propose an emotion lexicon construction method and features to address the problems above. Our main contributions are the following:

1. *Emotion lexicon construction.* We propose the weighted tweet frequency (weighted TwF) method, a simple and automatic way to build emotion lexicon lists directly from an annotated corpus without using other resources. The method will be useful for many countries where relevant resources are not available.
2. *Feature engineering.* We propose a set of fine-grained and language-specific features that improves the overall performance of machine learning based emotion classification in Korean Twitter texts.
3. *Resource and Dataset* Our study is unique because emotion analysis on Korean Twitter texts has rarely been addressed before. In addition, we built an annotated dataset, emotion lexicon sets, and other resources. Since finding related datasets and resources in Korean is difficult, we believe our work can contribute to future related studies.

The rest of this paper is organized as follows. Section 2 overviews related work, and in Section 3, we introduce our annotated dataset. In Section 4, we present our emotion lexicon construction method and in Section 5, we describe features designed for classification. We provide experimental results and analysis in Section 6 and conclude in Section 7.

## 2 Related Work

There have been extensive studies on sentiment analysis that classify expressions of sentiment into positive and negative emotions. In the last few years, researchers have started to explore finer granularity of emotion because simple division of polarity may not suffice in many real-world applications. There are two main approaches to emotion analysis, one is a lexicon based approach and the other is a machine learning based approach. The lexicon based approach utilizes a dictionary of words annotated with their emotional orientation and simply counts the words or aggregates the according values presented in texts. In contrast, the machine learning based approach performs classification using machine learning algorithms based on carefully designed features. Roberts et al. (2012) tried to classify seven emotions in the Twitter domain with binary support vector machine(SVM) classifiers, and Balabantaray et al. (2012) also used SVM classifiers with features including WordNet-Affect emotion lexicons.

A large number of existing emotion lexicon sets were built manually such as WordNet-Affect (Strapparava and Valitutti, 2004) and Linguistic Inquiry and Word Count (Pennebaker et al., 2001). Crowdsourcing is often utilized to obtain a large volume of human annotated lexicon sets such as the NRC Word-Emotion Association Lexicon (Mohammad and Turney, 2013). Non-English speaking countries like Korea have difficulties building emotion lexicons without human labor because existing lexicons and crowd-sourcing platforms are mostly available in English. To deal with the difficulties, one popular approach is to build lexicons upon other resources. For example, AffectNet (Cambria and Hussain, 2012) was constructed using ConceptNet(Liu and Singh, 2004) and WordNet-Affect (Strapparava and Valitutti, 2004). Another popular choice for building lexicon sets automatically is translating existing lexicon lists written in English. Those built by Remus et al. (2010) and Momtazi (2012) are examples. We also propose an automatic method that does not require lexical resources and translation.

Very few attempts have been made so far to analyze emotions in Korean text. Cho and Lee (2006) identified eight emotions in Korean song lyrics with manually annotated word emotion vectors. Lee et al.

(2013) classified Korean tweets into seven emotions and achieved 52% accuracy when using the multinomial naïve Bayes algorithm with morpheme features. The only publicly available Korean emotion lexicons we found were a set of 265 terms of nine emotion types, manually built by Rhee et al. (2008). Our work differs from the aforementioned Korean studies because we automatically construct larger emotion lexicon sets and introduce fine-grained features that are particularly attuned for Korean Twitter texts.

### 3 Korean Twitter Emotion Analysis (KTEA) Dataset

A Twitter dataset annotated by emotion types is essential in the machine learning based approach for the purpose of training. To build the corpus, we collected random Korean Twitter messages using Twitter streaming API. We removed tweets with RT, URL links, and replies. After the collection process, a corpus can be annotated either manually by human annotators or automatically by distant labels (Go et al., 2009; Wicaksono et al., ; Lee et al., 2013). In our case, we manually annotated the corpus. Each tweet was labeled by three annotators, producing three emotion labels per tweet. Consequently, we constructed a Korean Twitter Emotion Analysis (KTEA) dataset<sup>2</sup>, which contains 5,706 valid tweets labeled by seven types of emotions - Ekman’s six emotions and no emotion(neutral). Using the dataset, we constructed emotion lexicon sets as described in Section 4 and trained our machine learning algorithm as presented in Section 5.

## 4 Constructing Emotion Lexicons

### 4.1 Our Approach - Weighted Tweet Frequency

We built emotion lexicons automatically from the annotated corpus without using other lexical resources. For the construction, we utilized part of our KTEA dataset, which is the set of tweets, each of which was labeled as representing one of Ekman’s six emotion types (disregarding the neutral case) by at least one annotator. Table 1 shows the number of tweets we used per emotion for the purpose of lexicon construction.

Emotion	Number of Tweets
Happiness	770
Sadness	1377
Anger	903
Disgust	694
Surprise	475
Fear	228
Total	4447

Table 1: The number of tweets we used to generate emotion lexicons using weighted TwF approach

To generate emotion lexicons, we propose the weighted tweet Frequency (weighted TwF) method. First, we aggregated tweets of the same emotion label in one document ( $d$ ), producing six documents ( $D$ ) of tweets as a result. Using the six documents, we calculated the weighted TwF for each term ( $t$ ) that appeared in the documents. The weighted TwF is expressed in Equations 1, 2, and 3. Consequently, we generated six emotion lexicon lists, one list for each emotion type. Each lexicon has a weighted TwF value which shows the strength of the corresponding emotion, i.e., the higher the value is, the stronger the emotion is. The basic idea is similar to the concept of term frequency - inverse document frequency (TF-IDF)<sup>3</sup>, for which the occurrences of a term are counted and a penalty is given if the term appears in several documents. However, TF-IDF is not appropriate for our task because the structures of tweets are often highly ungrammatical, and there are many tweets with meaningless terms, which are sometimes excessively repeated in one tweet. In such cases, the meaningless terms produce high term frequency, which results in erroneous emotion lexicons. As illustrated in Figure 1, when term frequency (TF-IDF) is used, we can see some words (that are names in this example), such as 시우민 “Xiumin”, 성규 “Sung Kyu”, and 김민석 “Kim Min Seok”, ranked high in the happiness lexicon list. This is because there are few tweets that excessively repeat those names. Similar kinds of unstructured tweets are frequent in Twitter, and we can disregard such cases by using the tweet frequency defined in Equation 1. It counts the number of tweets instead as true emotion lexicons appear across many tweets, not in a few erroneous tweets.

<sup>2</sup>goo.gl/Gu0GNw

<sup>3</sup><https://en.wikipedia.org/wiki/Tf-idf>

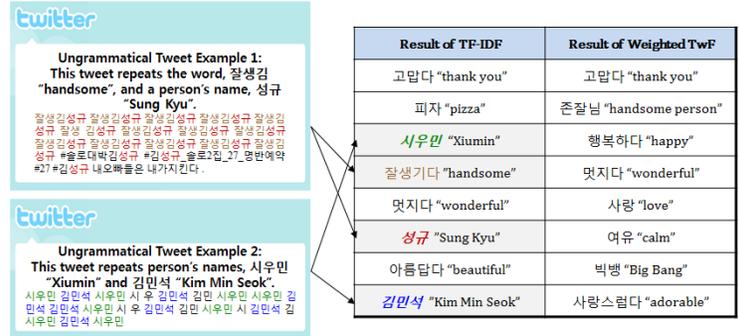


Figure 1: Example of top-ranked happiness lexicons generated from TF-IDF and weighted TwF

Another reason why TF-IDF is not suitable is the log term in IDF, which is trivial due to the small number of documents. Thus, we used a simple weighting scheme instead as in Equation 2. We set the weight to zero when a lexicon appeared in all the emotion documents in order to remove lexical items that appear very frequently but without any emotions, for example, 있다 "is", and 나 "I".

- $d$  : A document with tweets of same emotion
- $D$  : Total set of  $ds$
- $t$  : Target term
- $n$  : The number of  $ds$  where  $t$  appears

$$\begin{aligned} &\text{Normalized Tweet Frequency} \\ &= \frac{\text{Number of tweets in } d \text{ where } t \text{ appears}}{\text{Total number of terms in the } d} \end{aligned} \quad (1)$$

$$\text{Weight} = \begin{cases} \frac{1}{n} & n < |D| \\ 0 & n = |D| \end{cases} \quad (2)$$

$$\begin{aligned} &\text{weighted Tweet Frequency (weighted TwF)} \\ &= \text{Normalized Tweet Frequency} \times \text{Weight} \end{aligned} \quad (3)$$

Automatic methods of building emotion lexicons have been studied in many works. There are two widely used methods, namely, a thesaurus-based approach (Section 4.2) and a translation-based approach (Section 4.3).

#### 4.2 Thesaurus-Based Approach

The thesaurus-based method builds emotion lexicon lists using synonyms. Using a small set of emotion seed words, this method looks for synonyms using a

thesaurus and adds them to the emotion lexicon lists. Due to the lack of a large and representative Korean thesaurus, we combined various publicly available resources, namely, Dong-a's Prime dictionary<sup>4</sup>, Naver dictionary<sup>5</sup>, a Korean thesaurus<sup>6</sup>, and Wise-WordNet<sup>7</sup>. First, seed words – happiness, sadness, anger, disgust, surprise, and fear – were translated into Korean using Dong-a's Prime English-Korean Dictionary. Then, we extended the emotion lexicon sets to include derivatives and synonyms using various resources and thesauruses. Since the resources were not perfect, there were many erroneous synonyms. Thus, for the last step, we manually removed the unreasonable ones. The detailed procedure is summarized in Table 2.

#### 4.3 Translation-Based Approach

There are many lexical resources in English for emotion analysis. This method translates such resources to a specific language, in our case, Korean. Among many lexical resources, we chose WordNet-Affect (Strapparava and Valitutti, 2004) as it is one of the popular and typical emotion lexicon sets used in emotion analysis, and it is freely available. WordNet-Affect contains WordNet synonyms and is manually annotated by Ekman's six emotions. We translated the WordNet-Affect list using Google Translate<sup>8</sup>. We employed the Google service as it is the most widely used translator and its performance is known to be fairly accurate. However, there were

<sup>4</sup><http://www.dongapublishing.com/entry/index.html>  
<sup>5</sup>[dic.naver.com](http://dic.naver.com)  
<sup>6</sup><http://www.wordnet.co.kr/>  
<sup>7</sup>Software Research Laboratory, ETRI  
<sup>8</sup><https://translate.google.co.kr/>

Thesaurus-Based Approach
<b>Seed words</b> happiness, sadness, anger, disgust, surprise, fear (each seed word constructs according emotion lexicon list)
<b>Step 1.</b> Translate seed words to Korean using Dong-a’s Prime dictionary
<b>Step 2.</b> Add derivatives using NAVER dictionary
<b>Step 3.</b> Using Korean thesaurus, add synonyms of each word
<b>Step 4.</b> Using WiseWordNet, add primary synonyms of each word
<b>Step 5.</b> Leave only exclusive words for each emotion and remove duplicates within list
<b>Step 6.</b> Manually remove unreasonable or misleading emotion words

Table 2: Procedure of making emotion lexicons using thesaurus-based approach

some erroneous translations since the Korean translator is not perfect. Thus, we manually modified and removed problematic words and duplicates.

#### 4.4 Comparison

In this section, we explain the qualitative aspects of our lexicon construction method in comparison with other approaches. The advantages of our emotion lexicon sets built by weighted TwF approach are the following:

1. As the wordlist is constructed based on real Twitter messages, the method generates Twitter-specific lexicons that include slang, swear words, and ungrammatical words. *Example: 존잘님 “slang for handsome person”, 조아 “ungrammatical word for like”*
2. Our method discovers topics that are closely related to some particular emotions. *Example: 야자 “night school study” (sadness - many students feel sad when they are forced to study at night in school)*
3. It is possible to discover keywords that particularly appear in a specific time range. The method automatically updates the lexicons to include newly-coined words, which are essential for emotion analysis in Twitter domain. *Example: 빅뱅 “Big Bang” (happiness - a famous Korean singer Big Bang released a new album at the time we constructed the emotion lexicons)*

We show the effectiveness of our weighted TwF approach by comparing it with the popular

Approach	Weighted TwF	Thesaurus	Translation
Automatic?	O	O	O
Resource-free?	O	X(thesaurus)	X(translator)
No manual work?	O	X	X
Twitter-specific?	O	X	X

Table 3: Comparison of our weighted TwF approach with thesaurus-based and translation-based approaches

thesaurus-based and translation-based approaches. Table 3 compares the three approaches. These approaches can automatically generate emotion lexicons. To be specific, using the thesaurus-based approach, we are able to construct emotion wordlists easily and automatically by using only a small set of seed words. The translation-based approach also translates the existing emotion lexicons automatically using translators. However, the thesaurus-based approach is heavily dependent on lexical resources like dictionaries and thesauruses. A well-built thesaurus is not likely to be available in many non-English speaking countries. Additionally, translation-based approach requires a reliable translator. In comparison, our weighted TwF approach is based on statistics, which are independent of lexical resources and translators; thus, it would be very useful for under-resourced countries. Moreover, we observed that the thesaurus- and translation-based approaches generate a lot of erroneous words due to errors of resources and translators. Hence, manual removal of those words was necessary to achieve accurate results. In contrast, our approach generates lexicons with strength values that show how accurately the word may belong to an emotion type. Even though erroneous words are included in the list, they are likely to be ignored due to the low weighted TwF value. Lastly, our lexicon sets are particularly attuned to the Twitter domain; they include slang, jargon, ungrammatical words, and newly-coined words, whereas most other approaches do not.

## 5 Machine Learning with Fine-Grained Features

Our goal is to classify Korean Twitter messages according to one of the following six emotions, happiness, sadness, anger, disgust, surprise, and fear. We used a machine learning algorithm to classify each Twitter message represented by a feature vector. We

first explain features that we propose in this work and explain our machine learning classification.

### 5.1 Fine-Grained Features

Feature engineering is very important in machine learning. Features that have been traditionally used in emotion analysis are lexicons and punctuations. Positive and negative emoticons such as :) and :( have also been used in some research. However, more fine-grained and language-specific features are necessary to distinguish finer granularity of emotions. To come up with some effective features, we worked with the following ideas:

- Emoticons and symbols may express specific types of fine-grained emotions
- Some alphabet letters may convey emotions
- Exclamation words may appear in surprise messages
- Swear words may appear in angry messages

We explain how we designed the features according to the ideas we presented above with some examples.

**Fine-Grained Emoticons and Symbols** Emoticons and symbols are important in analyzing online language because many people express their feelings using them. We constructed a list of emoticons for each fine-grained emotion type that are used in Korea as well as general emoticons widely used in Eastern and Western countries. We constructed a dictionary of emoticons and symbols with the aid of various website articles. Moreover, we included Emojis which have become increasingly popular on Twitter since mobile devices adopted them. We sorted each emoticon and symbol into one of the six emotions using the explanations written in the websites. One interesting aspect of the dictionary is that it utilizes regular expressions to incorporate various mutations of emoticons. For example, Korean emoticons often use various or extended particles to represent the mouth of a face. In the case of a smiling face (^), people use various mutation of such emoticons such as (^\_^),(^\_\_\_\_^),(^.^),(^.^),(^3^). In other words, similar to languages, emoticons also have informal versions of similar patterns. Thus, we incorporated such common cases with regular expressions. Part

Emotion	Happiness	Sadness	Anger	Disgust	Surprise	Fear
Examples	=) \\^(^\\ \\ )^\\^ 	ㅍㅍ T((^\\ \\ )^T 	>(< o'ㅍ'o 	-ㅈ- 0ㅍ0 	\\(OoO)/! O_o 	~_~ -:- 

Figure 2: Example of fine-grained emoticon-emotion dictionary

Korean Letters	Meaning
ㅋ, ㅎ	Laughing with Happiness
ㅠ, ㅓ	Crying with Sadness
ㄷㄷ	Shaking with Fear
ㅂㄷㅂㄷ	Trembling with Anger

Table 4: List of Korean letters closely related to emotions

of the dictionary of emoticons and symbols is shown in Figure 2.

**Korean Emotion Letters** Language-specific feature are important in performing emotion analysis for a specific language. Koreans use certain Korean letters to show emotions, so we took certain letters into account that are listed in Table 4. ‘ㅋ’ and ‘ㅎ’ are often used to indicate laughter, while ‘ㅠ’ and ‘ㅓ’ indicate crying. Also, sequences of letters, such as ‘ㄷㄷ’ and ‘ㅂㄷㅂㄷ’, are often used to express fear and anger, respectively. We counted and normalized the number of such emotion letters and added them as a language-specific feature.

**Exclamations of Surprise** According to Merriam-Webster dictionary, the definition of an exclamation is “a sharp or sudden cry, a word, phrase, or sound that expresses a strong emotion<sup>9</sup>”. We assumed that exclamations are often used in tweets expressing surprise, such as 맵소사 “oh no”, 앓 “oh dear”, and 우와 “wow”. We searched various websites and collected examples to make a list of exclamations of surprise. As a result, we constructed a list of 45 surprise exclamation words. We then counted the number of occurrences of such words in tweets and added them as a feature.

**Swear Words** We observed that swear words are often used in angry tweets; therefore, we assumed that there occurrence is a strong clue to identify tweets expressing anger. We constructed our own list of Korean swear words by combining numerous related

<sup>9</sup><http://www.merriam-webster.com/dictionary/exclamation>

resources and websites. As a result, a list of 227 Korean swear words was built. For each tweet, we counted the number of occurrences of swear words and added them as a feature.

Consequently, we designed a feature vector based on the conventional features as well as the features we presented above. To sum up, we considered the following features for classification:

1. Emotion lexicons (weighted TwF)
2. Emotion lexicons (thesaurus+translation)
3. Punctuation (? ! ? . , ~)
4. Fine-grained emoticons and symbols
5. Korean emotion letters
6. Exclamations of surprise
7. Swear words

## 5.2 Machine Learning Based Classification

Before constructing a machine learning classifier, we applied the synthetic minority oversampling technique (SMOTE) (Chawla et al., 2002) to the training set, a well-known oversampling method, which is known to be more effective than the plain oversampling method with replication. We preferred an oversampling method to an undersampling method since our dataset is highly imbalanced, and undersampling removes too many instances. SMOTE generates synthetic instances of the minority class by choosing a random point for each line segment between randomly selected neighbors from  $k$  nearest minority neighbors. As a result of applying SMOTE to our training set, we could make a balanced dataset, which is favored for most machine learning algorithms. We compared several machine learning algorithms for classification, including support vector machine (SVM), multinomial logistic regression, random forest, J48, naive Bayes, and zeroR.

## 6 Experimental Results and Analysis

We performed experiments using WEKA<sup>10</sup> to evaluate 1) our weighted tweet frequency method and 2) the performance of machine learning based classification using the feature vector we engineered.

**Dataset** For training and testing the machine learning algorithms, we used 899 Twitter messages from our KTEA dataset, which contains tweets for which

three annotators all agreed on the emotion type, excluding neutral. We performed 5-cross validation.

**Performance Measure** We used precision, recall and F-measure to evaluate the classification performance for each emotion type. Also, the weighted average of each measure was computed to determine the overall performance of unbalanced test dataset.

**Weighted Tweet Frequency** We investigated the performance of our lexicon building method, weighted tweet Frequency, and compared it with the performance of the thesaurus- and translation-based methods. We found that the lexicons based on the thesaurus- and translation-based approaches suffer from low coverage due to the lack of reliable words produced by the Korean resources and translator. Therefore, we combined the lexicon lists produced by the thesaurus- and translation-based approaches to make a larger emotion lexicon list. In other words, we compared our approach (weighted TwF) against the combined approach (thesaurus+translation). The precision, recall, and F-measure of using SVM is shown in Figure 3. The F-measure of our approach is higher than that of the thesaurus+translation approach. The precision of the thesaurus+translation approach is relatively high due to the manual removal of erroneous words from the lists. However, its recall is very low because it does not contain Twitter-specific words. Furthermore, our approach, used together with the thesaurus+translation approach, achieves the best performance.

**Machine Learning Based Classification** First, we investigated the most appropriate machine learning algorithm for classification. We tested various machine learning algorithms: SVM, multinomial logistic regression, random forest, J48, naive Bayes, and zeroR. Figure 4 shows the results. SVM produced the best precision, recall, and F-measure compared to the others.

We conducted another experiment to evaluate how well the features we proposed improved the performance of SVM. As shown in Figure 5, the best performance was observed when all the features were combined and the overall F-measure was about 70%. Emotion lexicon and punctuation features achieved an F-measure of about 64%. Adding the exclamation of surprise feature improved the classification of the surprise emotion by a 12% F-measure. Further adding Korean emotion letters

<sup>10</sup><http://www.cs.waikato.ac.nz/ml/weka/>

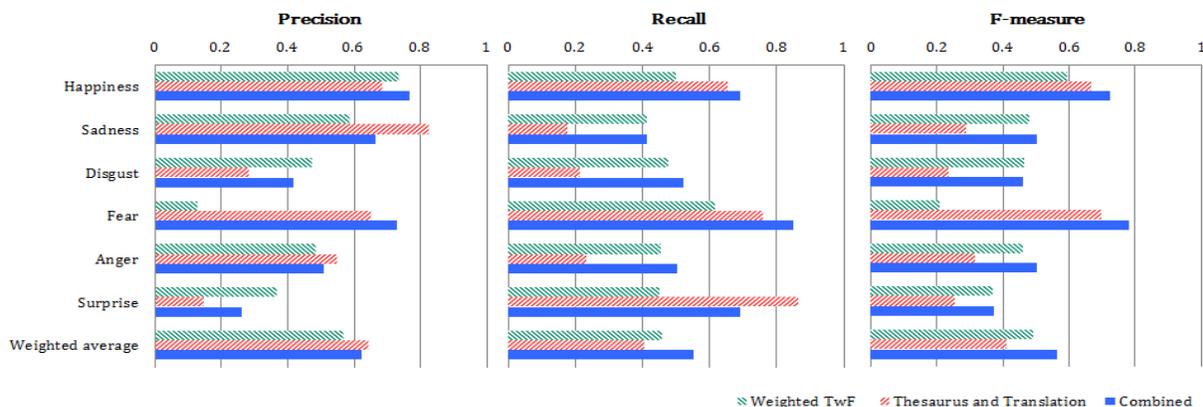


Figure 3: Precision, recall, and F-measure of using our weighted TwF, thesaurus+translation, and the two approaches combined. The best accuracy was observed when all the approaches were used.

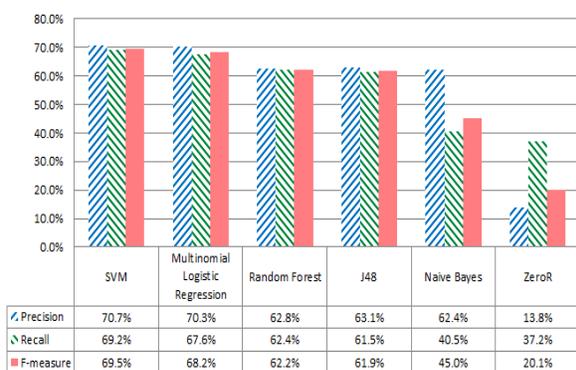


Figure 4: Precision, recall, and F-measure achieved by using different machine learning algorithms. SVM generated the best performance.

helped to classify sadness, increasing the classification score from 67% to 72.1%, while the value for fear increased from 76.3% to 79.5%. Moreover, combining emoticon and symbol feature particularly improved the classification for happiness increased from 73% to 76.3%. Lastly, we added the swear word feature. As expected, it increased the classification of anger from 54.4% to 56.5%. Overall, we found that our fine-grained features helped the analysis of fine-grained emotions, and we believe that improving the feature resources will further improve the overall performance.

## 7 Conclusion

We proposed a machine learning based classification method that sorts Korean Twitter messages into

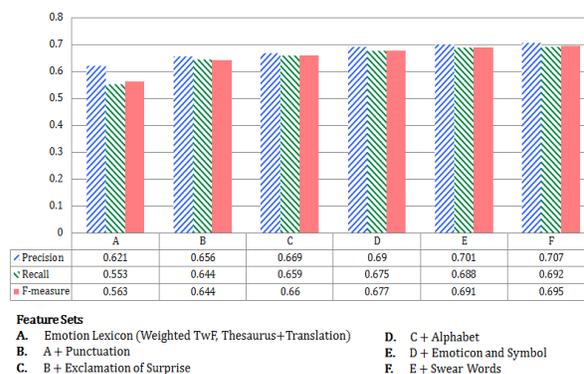


Figure 5: Precision, recall, and F-measure using combined set of features. Using all features achieved the best performance which is about 70% F-measure

six emotion types using carefully designed features. Emotion analysis research in under-resourced countries can benefit from our emotion lexicon building method as we automatically construct lexicons without any help from other resources and tools. In addition, we suggested several fine-grained features to improve classification performance. We believe that our research, the KTEA dataset, and resources represent a significant step forward in Korean Twitter emotion analysis studies, which have been rarely addressed before.

## Acknowledgment

This work was supported by the National Research Foundation of Korea(NRF) grant funded by the Korea government(MSIP) (No. 2010-0028631).

## References

- R. C. Balabantaray, Mudasir Mohammad, and Nibha Sharma. 2012. Multi-class twitter emotion classification: A new approach. *International Journal of Applied Information Systems*, 4(1):48–53.
- E. Y. Bann and J. J. Bryson. 2012. The conceptualisation of emotion qualia: Semantic clustering of emotional tweets. In *Proceedings of the 13th Neural Computation and Psychology Workshop*. World Scientific.
- Johan Bollen, Huina Mao, and Xiaojun Zeng. 2011. Twitter mood predicts the stock market. *Journal of Computational Science*, 2(1):1–8.
- Erik Cambria and Amir Hussain. 2012. *Sentic Computing: Techniques, Tools, and Applications*, volume 2. Springer Science & Business Media.
- Nitish V. Chawla, Kevin W. Bowyer, Lawrence O. Hall, and W. Philip Kegelmeyer. 2002. Smote: synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, 16:321–357.
- Young Hwan Cho and Kong Joo Lee. 2006. Automatic affect recognition using natural language processing techniques and manually built affect lexicon. *IEICE Transactions on Information and Systems*, 89(12):2964–2971.
- Nicholas A. Diakopoulos and David A. Shamma. 2010. Characterizing debate performance via aggregated twitter sentiment. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 1195–1198. ACM.
- Son Doan, Bao-Khanh Ho Vo, and Nigel Collier. 2012. An analysis of twitter messages in the 2011 tohoku earthquake. In *Electronic Healthcare*, volume 91, pages 58–66. Springer.
- Paul Ekman. 1992. An argument for basic emotions. *Cognition & Emotion*, 6(3-4):169–200.
- Alec Go, Richa Bhayani, and Lei Huang. 2009. Twitter sentiment classification using distant supervision. *CS224N Project Report, Stanford*, 1:12.
- Jeongin Kim, Dongjin Choi, Myunggwon Hwang, and Pankoo Kim. 2014. Analysis on smartphone related twitter reviews by using opinion mining techniques. In *Advanced Approaches to Intelligent Information and Database Systems*, pages 205–212. Springer.
- Cheolseong Lee, Donghee Choi, Seongsoon Kim, and Jaewoo Kang. 2013. Classification and analysis of emotion in korean microblog texts. *Journal of Korean Institute of Information Scientists and Engineers: Databases*, 40(3):159–167.
- Hugo Liu and Push Singh. 2004. Conceptnet — a practical commonsense reasoning tool-kit. *Journal of BT Technology*, 22(4):211–226.
- Saif M. Mohammad and Peter D. Turney. 2013. Crowdsourcing a word–emotion association lexicon. *Computational Intelligence*, 29(3):436–465.
- Saeedeh Momtazi. 2012. Fine-grained german sentiment analysis on social media. In *Proceedings of the 8th International Conference on Language Resources and Evaluation*. European Language Resources Association (ELRA).
- Brendan O’Connor, Ramnath Balasubramanyan, Bryan R Routledge, and Noah A Smith. 2010. From tweets to polls: Linking text sentiment to public opinion time series. *Proceedings of the International Conference on Web and Social Media*, 11(122-129):1–2.
- James W. Pennebaker, Martha E. Francis, and Roger J. Booth. 2001. Linguistic inquiry and word count: Liwc 2001. *Mahway: Lawrence Erlbaum Associates*, 71:2001.
- Robert Remus, Uwe Quasthoff, and Gerhard Heyer. 2010. Sentiws - a publicly available german-language resource for sentiment analysis. In *Proceedings of the 7th International Conference on Language Resources and Evaluation*. European Language Resources Association (ELRA).
- June Woong Rhee, Hyun Joo Song, Eun Kyung Na, and Hyun Suk Kim. 2008. Classification of emotion terms in korean. *Korean Journal of Journalism and Communication Studies*, 52(1):85–116.
- Kirk Roberts, Michael A. Roach, Joseph Johnson, Josh Guthrie, and Sanda M. Harabagiu. 2012. Empatweet: Annotating and detecting emotions on twitter. In *Proceedings of the 8th International Conference on Language Resources and Evaluation*. European Language Resources Association (ELRA).
- Carlo Strapparava and Alessandro Valitutti. 2004. Wordnet affect: an affective extension of wordnet. In *Proceedings of the 4th International Conference on Language Resources and Evaluation*, volume 4, pages 1083–1086.
- Alfan F. Wicaksono, Clara Vania, Distiawan T. Bayu, and Mirna Adriani. Automatically building a corpus for sentiment analysis on indonesian tweets. *28th Pacific Asia Conference on Language, Information and Computing*.

# Chinese word segmentation based on analogy and majority voting

Zongrong Zheng    Yi Wang    Yves Lepage

Graduate School of Information, Production and Systems

Waseda University

2-7 Hibikino, Wakamatsu-ku, Kitakyushu, Fukuoka 808-0135, Japan

{z zr0427@toki., yiwang@akane., yves.lepage@}waseda.jp

## Abstract

This paper proposes a new method of Chinese word segmentation based on proportional analogy and majority voting. First, we introduce an analogy-based method for solving the word segmentation problem. Second, we show how to use majority voting to make the decision on where to segment. The preliminary results show that this approach compares well with other segmenters reported in previous studies. As an important and original feature, our method does not need any pretraining or lexical knowledge.

## 1 Introduction

Words are usually considered a basic unit in natural language processing (NLP) studies. As natural language texts are continuous sequences of characters, it is generally agreed that word segmentation is the initial step of NLP. The performance of the best Chinese segmenters for F-score has reached 95%, as reported in the second SIGHAN Chinese segmentation bakeoff (Emerson, 2005). These best existing methods rely on massive training data.

How to utilize as much information as possible from the training corpus to adapt a segmentation system towards a segmentation standard has been the main issue (Kit et al., 2005). Most of existing methods can be roughly classified as either dictionary-based or statistical-based methods.

Dictionary-based methods usually rely on large-scale lexicons and are built upon a few basic "mechanical" segmentation methods based on string

matching. Without a large, comprehensive dictionary, the success of such methods degrade.

Statistical-based methods consider the segmentation problem as a classification problem on characters and usually involve complicated language models trained on large-scale corpora.

All of these methods require pre-training data and prior lexical knowledge. All current methods assume comprehensive lexical knowledge. How to model human cognition and acquisition it to segment words efficiently without using knowledge of wordhood is still a challenge in CWS (Huang et al., 2007).

After this introduction, we shall introduce the notion of proportional analogy in section 2 on which our proposal relies. In section 3, we shall describe the main idea of our new method for CWS using proportional analogy. Section 4 shall present the details of our implementation of our method. Section 5 shall detail some experiments done to evaluate our method with other state-of-the-art methods.

## 2 Proportional Analogy

Analogy has shown great potential in natural language processing, like machine translation (Lepage et al., 2005) and semantic relations (Turney et al., 2005). A proportional analogy is a relationship between four objects, noted  $A : B :: C : D$  in its general form (Lepage et al., 2005). On numbers we have:

$$\frac{5}{15} = \frac{10}{30} \quad \text{also written as an analogy } 5 : 15 :: 10 : 30$$

By using words, sequences of words or sentences instead of numbers, we get proportional analogies

between words, sequences of words or sentences. For instance, the following example is a true analogy between sequences of words:

*I walked : to walk :: I laughed : to laugh*

We use the algorithm proposed by Lepage (1998) for the resolution of analogical equations. This algorithm is based on the formalization of proportional analogies shown in formula (1) (Lepage, 2004).

$$A : B :: C : D \Leftrightarrow \begin{cases} |A|_a - |B|_a = |C|_a - |D|_a, \forall a \\ dist(A, B) = dist(C, D) \\ dist(A, C) = dist(B, D) \end{cases} \quad (1)$$

Here,  $a$  is a character, whatever the writing system, and  $A, B, C$  and  $D$  are strings of characters.  $|A|_a$  stands for the number of occurrences of character  $a$  in the string of characters  $A$  and  $dist(A, B)$  stands for the edit distance between strings  $A$  and  $B$  which only considering insertions and deletions only as edit operations. The input of this algorithm is three strings of characters, words, sequences of words or sentences. Its output is a string of characters in analogy with the input. The following is an example applying this algorithm in Chinese:

我爱吃饭 : 我爱喝水 :: 你爱吃饭 : x  
x = 你爱喝水

### 3 A New Method for CWS using proportional analogy

We propose a new Chinese word segmentation method based on proportional analogies. Crucially, we no longer need any pre-processing phase (training) or lexical knowledge (dictionary). The following gives the basic idea of the method. We are inspired by the example-based machine translation system proposed by Lepage et al. (2005).

Let us suppose that we have a corpus of sentences in their usual unsegmented form and their segmented form. We call it the training corpus. A line in such a training corpus may look like:

unsegmented form # segmented form  
迈向充满希望的新世纪#迈向\_\_充  
满\_\_希望\_\_的\_\_新\_\_世纪

Let  $D$  be an input sentence to be segmented into segmented sentence  $\tilde{D}$ .

- (i) We build all analogical equations  $A_i : B_j :: x : D$  with the input sentence  $D$  and with all pairs of sub-strings  $(A_i, B_j)$  from the unsegmented part of the training corpus. According to formula (1), not all analogical equations have a solution. In order to get more analogical solutions and reduce time in solving analogical equations, we only consider sub-strings  $A_i$  and  $B_i$  which are more similar to  $D$  than a given threshold;
- (ii) We gather all the solutions  $x$  of the previous analogical equations and only keep the solutions, named  $C_{i,j}$ , which belong to the training corpus. As it is easy to map from unsegmented part to segmented part for any sub-strings in training corpus, for each  $C_{i,j}$ ,  $A_i$  and  $B_i$ , we easily retrieve their corresponding segmented forms  $\widetilde{C}_{i,j}$ ,  $\widetilde{A}_i$  and  $\widetilde{B}_i$  in the segmented part of the training corpus;
- (iii) We then form all possible analogical equations with all pairs  $(A_i, B_j, C_{i,j})$ :

$$\widetilde{A}_i : \widetilde{B}_i :: \widetilde{C}_{i,j} : y$$

We output the solutions  $y = \widetilde{D}_{i,j}$  of all these analogical equations. They are hypotheses of segmentation for  $D$ . We record the number of times of each hypotheses. Recall that different analogical equations may generate identical solutions.

Figure 1 gives a simple example to illustrate the basic work flow of the method described above.

### 4 A CWS system using proportional analogy

In this section we describe the details of our implementation of the analogy-based word segmentation method. The key point in our method is to generate as precise proportional analogies as possible. These solutions of proportional analogy are the segmented results of input sentences. As not all of these solutions are exactly correct, we will consider them



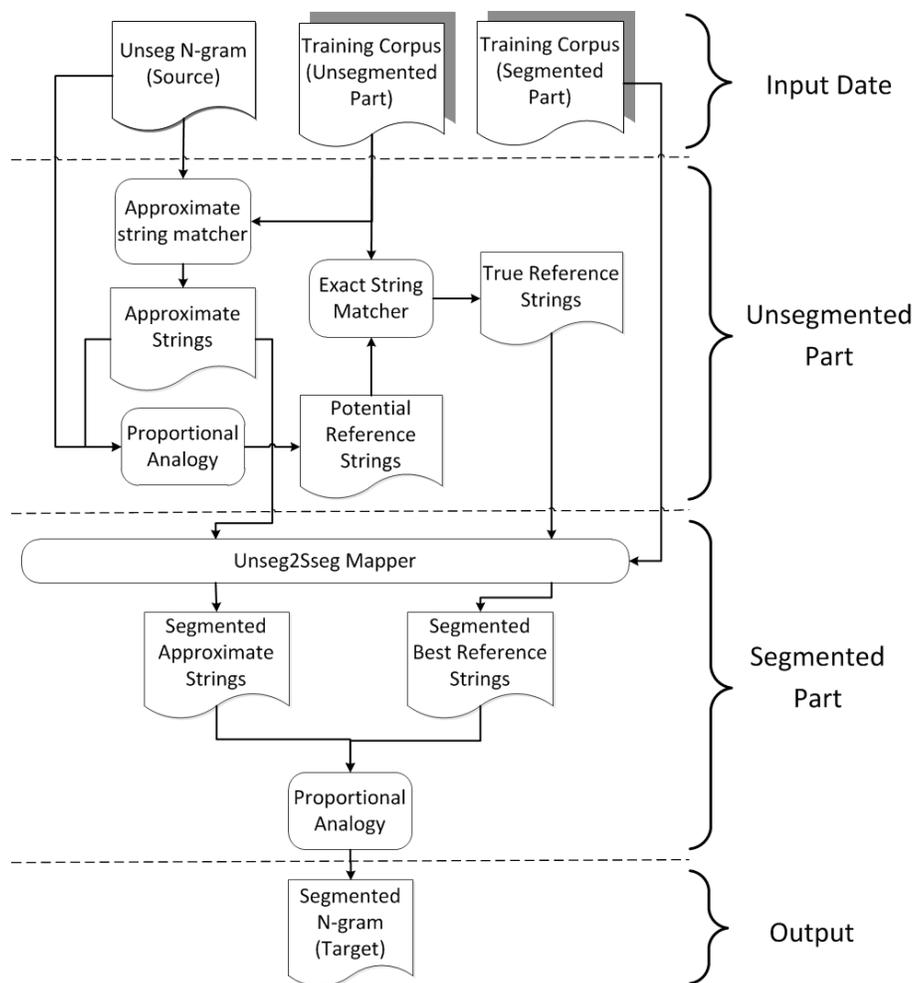


Figure 2: Work flow of generating segmented reference of n-grams in our system

	c <sub>1</sub>	D <sub>1</sub>	c <sub>2</sub>	D <sub>2</sub>	c <sub>3</sub>	D <sub>3</sub>	c <sub>4</sub>	D <sub>4</sub>	c <sub>5</sub>	D <sub>5</sub>	c <sub>6</sub>	D <sub>6</sub>	c <sub>7</sub>	D <sub>7</sub>	c <sub>8</sub>	D <sub>8</sub>	c <sub>9</sub>
# of occurrence	人		类		社		会		前		进		的		航		船
9	人		类	┌	社		会	┌	前								
3			类	┌	社		会	┌	前		进						
1					社		会		前	┌	进	┌	的				
11					社		会		前		进	┌	的				
37					社		会	┌	前		进	┌	的				
4									前		进	┌	的	┌	航		船
1									前		进		的		航		船
for seg (┌)		0		12		0		49		1		53		4		0	
against seg		9		0		61		12		56		1		1		5	
segmentation result	人		类	┌	社		会	┌	前		进	┌	的	┌	航		船

Figure 3: An example of recombination of segmentation hypotheses of n-grams using majority voting

	PKU
Word tokens	104372
Word types	13148
OOV words tokens	6006
OOV words types	2863
Character tokens	172733
Character types	2934
OOV character tokens	372
OOV character types	92

Table 1: Corpus details of PKU test set.

up the votes in favor and against segmentation and output the final results according to the vote results.

## 5 Experiments

### 5.1 Data and Evaluation

To evaluate the effectiveness of our proposed method, we conduct experiments on a widely used Chinese word-segmented corpora, namely PKU, from the second SIGHAN international Chinese word segmentation bakeoff (Emerson, 2005). The training set and the test set are publicly available from the official website<sup>2</sup>. Table 1 shows some statistics on the data sets. All evaluation results in this paper are tested by the official scoring script, also downloaded from the official website.

The segmentation accuracy is evaluated by test recall (R), test precision (P) and balanced F-score, as defines in Equation (2), (3) and (4).

$$R = \frac{\text{number of correctly segmented words}}{\text{total number of words in gold standard segmentation}} \quad (2)$$

$$P = \frac{\text{number of correctly segmented words}}{\text{total number of words in segmentation result}} \quad (3)$$

$$F = \frac{2 \times P \times R}{P + R} \quad (4)$$

Our experiments follow the closed track. It means that no extra resource other than training corpora is used.

<sup>2</sup><http://www.sighan.org/bakeoff2005/>

Models	PKU Corpus				
	P	R	F	$R_{OOV}$	$R_{iv}$
baseline	84.3	90.7	87.4	6.9	95.8
Best05 closed-set	95.4	94.6	95.0	78.7	95.6
<b>This work</b> (closed-set)	90.9	89.9	90.4	60.7	91.6

Table 3: Performance of our system on the SIGHAN 2005 data set. Best05 refers to the best closed-set results in SIGHAN 2005 bakeoff.

### 5.2 Effects of Length of $n$ -grams and Edit Distance

As discussed in section 4, long sentences are easier to miss hypotheses of segmentation. So the length of  $n$ -grams will influence the segmentation results. Moreover, the larger edit distance is used, the more similar sub-strings would be retrieved. To measure it, we conduct experiments using different length of  $n$ -grams and different edit distance.

According to our majority voting method, we would consider a position is not segmented if no segmentation hypothesis votes for it. The results in Table 2 shows that this data sparse problem is more serious when we used larger length of  $n$ -grams.

### 5.3 Results

We set length of  $n$ -grams to 3 and edit distance to 2 for approximate string match to perform our experiments. Table 3 shows our empirical results on the data set. Our system achieve a significantly better results than the baseline.  $R_{iv}$  score shows that our method performs well on in vocabulary (IV) word recognition. Simultaneously, the  $R_{OOV}$  score shows that our method has certain ability to deal with out-of-vocabulary (OOV) word and guess their form. Compared with best result (Tseng et al., 2005) in SIGHAN 2005, our result still has a lot of room for improvement. But as a original method which do not need any pre-training or lexical knowledge, our method has a great potential in CWS.

## 6 Conclusion

In this paper, we presented an approach to Chinese word segmentation based on proportional analogy and majority voting to make decision on where to segment. Our approach achieves a desirable accuracy, when evaluated on the corpus of the close track of SIGHAN 2005 and shows an excellent perfor-

# of $n$ -grams	Edit Distance	Word Count	P	R	F
6	3	79828	85.5	65.4	74.1
5	3	95079	90.0	82.0	85.8
4	2	99103	90.8	86.2	88.4
3	2	103186	90.9	89.9	90.4

Table 2: Performance of our method with different length of  $n$ -grams and edit distance.

mance in word identification. As an important and original feature, our method does not need any pre-training or lexical knowledge.

## References

- Thomas Emerson. The second international chinese word segmentation bakeoff. In *Proceedings of the fourth SIGHAN workshop on Chinese language Processing*, volume 133, 2005.
- Chu-Ren Huang, Petr Šimon, Shu-Kai Hsieh, and Laurent Prévot. Rethinking chinese word segmentation: tokenization, character classification, or wordbreak identification. In *Proceedings of the 45th annual meeting of the ACL on interactive poster and demonstration sessions*, pages 69–72. Association for Computational Linguistics, 2007.
- Chunyu Kit and Xiaoyue Liu. An example-based chinese word segmentation system for cwsb-2. In *Proceedings of the Fourth SIGHAN Workshop on Chinese Language Processing*, pages 146–149, 2005.
- Yves Lepage. Solving analogies on words: an algorithm. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics-Volume 1*, pages 728–734. Association for Computational Linguistics, 1998.
- Yves Lepage. Analogy and formal languages. *Electronic notes in theoretical computer science*, 53:180–191, 2004.
- Yves Lepage and Etienne Denoual. Purest ever example-based machine translation: Detailed presentation and assessment. *Machine Translation*, 19(3-4):251–282, 2005.
- Huihsin Tseng, Pichuan Chang, Galen Andrew, Daniel Jurafsky, and Christopher Manning. A conditional random field word segmenter for sighan bakeoff 2005. In *Proceedings of the fourth SIGHAN workshop on Chinese language Processing*, volume 171, 2005.
- Peter D Turney and Michael L Littman. Corpus-based learning of analogies and semantic relations. *Machine Learning*, 60(1-3):251–278, 2005.

## Enhancing Root Extractors Using Light Stemmers

**Mahmoud El-Defrawy**  
College of Computing  
and Information Technology  
AAST, Alexandria, Egypt  
eldefrawy.mahmoud@yahoo.com

**Yasser El-Sonbaty**  
College of Computing  
and Information Technology  
AAST, Alexandria, Egypt  
yasser@aast.edu

**Nahla A. Belal**  
College of Computing  
and Information Technology  
AAST, Alexandria, Egypt  
nahlabelal@aast.edu

### Abstract

The rise of Natural Language Processing (NLP) opened new possibilities for various applications that were not applicable before. A morphological-rich language such as Arabic introduces a set of features, such as roots, that would assist the progress of NLP. Many tools were developed to capture the process of root extraction (*stemming*). Stemmers have improved many NLP tasks without explicit knowledge about its stemming accuracy. In this paper, a study is conducted to evaluate various Arabic stemmers. The study is done as a series of comparisons using a manually annotated dataset, which shows the efficiency of Arabic stemmers, and points out potential improvements to existing stemmers. The paper also presents enhanced root extractors by using light stemmers as a preprocessing phase.

### 1 Introduction

Natural Languages (NLs) are the medium that allow two or more parties to communicate and interact. Linguistics have captured NLs as a set of sophisticated rules that describe the usage of a language.

The merger between linguistics and computer science began to formalize into Natural Language Processing (NLP) in mid 1950s (Nadkarni et al., 2011). Machine Translation (MT) (Hutchins, 2004) was one of the first tasks of NLP. MT takes one language as an input then predicts the output in another language. Linguistic complexity limited the development of MT, and other NLP tasks (Nadkarni et al., 2011).

There exists various NLP tasks. For example, Text Summarization (Jing and McKeown, 2000)(Nenkova, 2005), Part of Speech Tagging (POST) (Habash et al., 2009), word segmentation (Monroe et al., 2014), sentiment analysis (Oraby et al., 2013b)(Oraby et al., 2013a), and many more tasks. Each task has a specific goal which can be achieved by utilizing another NLP task. For example, sentiment analysis utilizes stemming algorithms (Oraby et al., 2013a). Many NLP tasks are a part of more complex tasks.

Text plays a central role in NLP and can be found in different forms, such as simple text or extracted from images (Fathalla et al., 2007), this increases text resources and hence increases the need for more concise text forms.

Stemming analysis is essential for many complex tasks. Stemming is a way of reducing a given word into a concise representation while preserving most of its linguistic features (Ryding, 2005). Arabic language is highly supportive for stemming analysis. Arabic language is a derivative language, where words are constructed from basic forms called roots (Ryding, 2005). Stemming for the Arabic language is the process of deriving back the root of a given word. Some stemmers derive multiple roots for a single word, hence, various techniques were used to disambiguate multiple roots. For example, utilizing a words context was used in the technique Context-Based Arabic Stemmer, CBAS, proposed in (El-Defrawy et al., 2015). Arabic stemmers are utilized for many tasks, such as sentiment analysis (Saleh and El-Sonbaty, 2007)(Oraby et al., 2013b)(Oraby et al., 2013a), question answering (Ezzeldin et al.,

2013), and Information Retrieval (IR) (Aljlayl and Frieder, 2002a)(Larkey et al., 2002)(Taghva et al., 2005).

In this paper, a study is conducted to analyze and compare different Arabic stemmers from different perspectives, using a manually annotated dataset. Moreover, the paper presents an enhanced version of root extractors using light stemmers for preprocessing. The paper is organized as follows, section 2 presents a concise introduction of Arabic morphology, which gives the basic intuition of Arabic stemming analysis. Section 3 discusses various techniques and strategies used to develop Arabic stemmers, followed by a detailed comparison and evaluation of existing Arabic stemmers in section 4. Finally, a conclusion is presented in section 5.

## 2 Background

Understanding the linguistic theory about derivational analysis provides intuitive reasoning behind different choices Arabic stemmers would do, and highlights their capabilities, strengths, and weaknesses. This section outlines the theory behind Arabic morphology and the main challenges associated with it. Arabic morphology is the study of a words construction, a new word generated from a root (Ryding, 2005). A new word is generated by changing its root. For example, the word ناجح (nāḡḡḥ, means "Successful") is generated from root ح ج ن (nūn ḡym ḡāḡ, means "Success") by adding ا (ʾlf) in the middle.

Arabic morphology uses a set of templates which are called patterns. Patterns are accurately defining possible changes to a root to generate a word. Pattern is a sequence of letters that captures the structure of the new word (Ryding, 2005). There are two types of letters that constitute the pattern. The first is a generic set of letters ف (fāʾ) ع (ʿyn) ل (lām) that represent a roots letters. The second type is augmented letters, which represents possible additions. Augmented letters are represented by themselves in the pattern, such as the pattern فاعل (fāʿl, means "Actor of the verb") which used to generate the word ناجح (nāḡḡḥ, means "Successful"), the augmented letter ا (ʾlf) is represented by itself in the pattern. There are ten letters which can be used as augmented letters. It has been collected in

the word سألتمونيها (sʾltmūnyhā). The root-pattern system (Ryding, 2005) starts by substituting a roots letter into the patterns generic letters, where a new word is generated. There are some cases where some additional modifications are required, commonly due to grammatical rules and letters compatibility, which is not captured neither in the root nor the pattern.

### 2.1 Vocalization and Mutation

Vocalization is a words letter transformed from one form to another, mostly due to grammatical or phonological rules. Vocalization defines the rules of handling weak letters, and Hamza (ء, Arabic letter) in different situations. For example, the root ق و ل (qāf wāw lām, means "Saying") transforms to the word قال (qāl, means "Said") depending on the tense of the sentence, where the weak letter و (wāw) is transformed into the weak letter ا (ʾlf). Similarity, mutation follows a similar behavior but for a different reason. For example, the word اضطراب (ʾḡḡrāb, means "Disturbance") transforms to اضطراب (ʾḡḡrāb, means "Disturbance"), due to phonological incompatibility between ض (ḡḡḡ) and ت (tāʾ) which results ت (tāʾ) being transformed to ط (ḡāʾ). Vocalization and mutation are common challenges that face constructing Arabic stemmers.

### 2.2 Prefixes and Suffixes Addition

Prefixes and suffixes addition (Ryding, 2005) is a categorization to a type of augmented letters. It describes the augmented letters additions to the front or to the end of the root. Patterns can be defined to represent such additions. However, some letters can be added to the front or the end which are not part of augmented letters. For example, the word كتابك (ktābk, means "your book"), the letter ك (kāf) at the end is added as an indication to ownership. It is not part of the word itself and it is also not part of augmented letters. Defining new patterns with prefixes and suffixes attached not only would increase the number of patterns, but it will also break the augmented letters rule, which is preferable to define in a separate process to avoid breaking the Arabic linguistic model of having ten augmented letters سألتمونيها (sʾltmūnyhā).

### 2.3 Stopwords

Arabic language defines a set of words that have a special meaning, such as **في** (fy, means "In") and **من** (mn, means "From"). Such words do not follow root-pattern substitution and commonly have some static forms (Ryding, 2005). It is part of Arabic morphology to identify such words and skip it.

### 2.4 Diacritics

Diacritics (Ryding, 2005) are part of Arabic words semantics. It encapsulates a set of invaluable features capturing grammatical, morphological, and phonological information. Diacritics are annotations on individual letters of a word, but it is optional (Pasha et al., 2014). Most of the Arabic readers depend on their intuition to capture such information. Many Arabic resources do not include Diacritics, such as newspapers and non-linguistic books, which creates another challenge for Stemming algorithms.

Stemming is the reverse process of derivational analysis, where for a given word a root needs to be extracted. For Arabic speakers, stemming is fairly simple even with missing information. They are capable of deducing the correct root. But, for computational devices, it is a highly complex process. Even with a complete representation of Arabic morphology, the missing information of input, such as diacritics, may lead to a set of possible results. The challenges presented above enforce scientists to make different assumptions when constructing stemmers to find an appropriate balance between correct and incorrect results.

## 3 Related Work

Various stemmers were developed to utilize Arabic morphological features. Each stemmer developed some mechanism to extract these features. In this section, we explore major stemmers, and their techniques.

### 3.1 Khoja Stemmer

Khoja stemmer (Khoja and Garside, 1999) starts by removing diacritics, punctuation, and non-characters of the input word. The word then follows a set of predefined paths, such as a decision tree. The

paths are initially based on the words length then a series of prefixes and suffixes removals are defined. The resulting word gets matched with a set of predefined patterns. The matching process is highly complex, since it involves an additional set of linguistic rules. Finally, the extracted root gets validated against a set roots dictionary, then the process is terminated if the root is correct. In case the extracted root is incorrect, the stemmer continues searching for other root possibilities. The process is terminated when it reaches the first correct root, or after exhaustive search without finding a root, and it is then marked as an un-stemmed word. The number of used patterns is relatively small, indicating that Khoja stemmer is intensively dependent on prefixes and suffixes removal. Khoja stemmer is one of the closest simulations to the manual root extraction. The decisions made by Khoja stemmer are static, where it has a linguistic justification for each decision. But, it does not capture the dynamics of the language and it does not explore all linguistic possibilities. Khoja stemmer turns out to be a powerful tool (AlSughaiyer and AlKharashi, 2004). However, it does not involve other important cases, such as mutation, and the complexity of its decisions makes it challenging to update.

### 3.2 Sebawi Stemmer

Sebawi (Darwish, 2002) uses a different approach to build an Arabic stemmer. It utilizes a set of word-root pairs to deduce Arabic patterns, prefixes, and suffixes. The knowledge of the word and root makes it possible to segment the word into three parts, prefix, suffix, and stem (infix). The stems characters are then aligned with roots characters to formulate a pattern. The deduced patterns vary from linguistically defined patterns (Darwish, 2002). For example, the word **مكتوب** (mktūb, means "Written") when aligned with its root **ك ت ب** (kāf tā' bā', "Writing") would result **م** (mym) as prefix and pattern **فَعُول** (f'ūl) instead of the actual pattern **مَفْعُول** (mf'ūl). Sebawi (Darwish, 2002) keeps track of prefixes, suffixes, and deduced pattern counts, which will be used in the stemming analysis. In the root extraction process, an input word is entered, the stemmer searches for possible matches in the deduced patterns, when it matches prefix, suffix, and pattern, a root is extracted. However, there is a potential that the input word would

match two or more patterns; Sebawi utilizes the frequencies computed from the pattern deduction to associate a score with each possible match based on the conditional probability of prefix, suffix, and deduced pattern. Finally, the resulting roots are compared to an Arabic root dictionary to validate their existence (Darwish, 2002). The deduction of pattern removes the need for manually enumerating them. However, the deduced patterns are different from the linguistic patterns, deduced patterns introduce new patterns not previously used and in many cases deduced frequencies will not reflect the actual linguistic frequencies.

### 3.3 Light10 Stemmer

Light stemming is a less complex version of stemming analysis (AlSughaiyer and AlKharashi, 2004). Light stemmers are more concerned with removing the prefix and suffix of a word (AlSughaiyer and AlKharashi, 2004). Aljlayl and Frieder (2002b) construct a light stemmer to show that light stemming has a higher potential than root extraction with respect to Information Retrieval (IR). Larkey et al. (2002) conducted a similar study by constructing a set of light stemmers and comparing them with Khoja stemmer. Both types of stemming analysis showed improvement in IR (Larkey et al., 2002). However, the Light10 outperforms various stemmers in IR and it is widely used in IR (Larkey et al., 2007). Light10 is a fast and straightforward algorithm. It starts by removing punctuation, diacritics, and non-Arabic letters. It mainly normalizes the Hamza with all of its variations to ʾ (ʾlf). Then it starts by removing prefixes according to a set of constraints.

### 3.4 ISRI Stemmer

ISRI stemmer (Taghva et al., 2005) is another simulation for the linguistic process similar to Khoja stemmer (Khoja and Garside, 1999). It starts by normalizing the input word, removing diacritics, and non-related Arabic characters. The key in normalization is unifying the different forms of Hamza to ʾ (ʾlf) which differs from Khoja stemmer (Khoja and Garside, 1999). The normalized word then follows a series of decisions to remove possible prefixes that is three, or less characters, and then map it to a group of patterns according to its length. ISRI

searches for possible matches within a groups patterns, if there is no match; it starts by removing possible suffixes. The stemming process should be stopped when the remaining length of the input word is three or less characters. Another key difference from Khoja stemmer is that ISRI does not validate roots against any type of dictionaries. ISRI is more oriented towards finding the minimal representation of an input word which can be used for information retrieval. The lack of dictionary has some side effects, such as the extracted roots are not necessarily correct, the root could be a meaningless set of characters. Roots would be unreliable for further processing, specially for linguistic based tasks.

### 3.5 Tashaphyne Stemmer

Tashaphyne is a light weight Arabic stemmer (Zerrouki, 2010). It uses similar approach to ISRI stemmer. Since, it searches for the minimum representation of an Arabic word (Zerrouki, 2010). But, It is not as greedy as ISRI stemmer. It starts by removing non-related letters in the root extraction process, such as diacritics. It uses two lists of prefixes and suffixes to segment a given word. Tashaphyne provides both a light stem or a root to the input word.

### 3.6 ElixirFM Morphological Analyzer

ElixirFM (Smrž, 2007) is a functional morphological analyzer that utilizes syntactic features to distinguish a words sense (Smrž, 2007). Arabic Grammar and Morphology are highly correlated (Ryding, 2005). Many of the prefixes and suffixes additions have grammatical justification, which contributes to the formulation of patterns, such as pronoun additions. ElixirFM uses such correlation to improve the root extraction process; it uses Prague Arabic Dependency Treebank (PADT) (Smrč et al., 2008) to provide annotated syntactic features associated with Buckwalter stem dictionary (Buckwalter, 2002) for additional morphological knowledge. ElixirFM also handles many cases, such as mutation, using orthographical and phonological rules. The ElixirFM generates all possible roots and associates all deduced features (reasons) to distinguish word senses. It also provides additional options, such as inflecting words in various forms. ElixirFM provides various levels of analysis, such as resolving words with or without tokenization.

### 3.7 MADAMIRA Morphological Analyzer

MADAMIRA (Pasha et al., 2014) is a morphological analyzer that provides a set of valuable features including stemming. MADAMIRA (Pasha et al., 2014) is composed of two sub-tools, MADA (Habash et al., 2009) and AMIRA (Diab et al., 2007). MADA annotates the input word with every possible morphological feature, such as diacritics and lemma (Habash et al., 2009). MADA is capable of predicting 19 morphological features by using 14 distinct Support Vector Machine (SVM) and N-gram language model to predict the other 5 features (Habash et al., 2009). AMIRA (Diab et al., 2007) includes a word Tokenizer, POST, and Base Phrase Chucker (BPC), where some tasks intersect with MADA. AMIRA uses a machine learning approach (SVM) for its predictions. AMIRA analysis is not as deep as MADA with respect to the intersected tasks which makes AMIRA relatively faster (Pasha et al., 2014). The merger extends both tools (Pasha et al., 2014). It is a dynamic tool that provides a set of valuable features to other tasks, such as Machine Translation (MT) and Named Entity Recognition (NER) (Pasha et al., 2014). MADAMIRA provides a light stemming analysis feature where it removes prefix and suffix from a word. It is a powerful tool that captures underlying data dynamics. However, it is dependent on the data quality and the learning features.

## 4 Stemmers Evaluation and Enhancement

Arabic stemmers have been evaluated using a standard IR test, due to the lack of existing stemmed datasets (Smirnov, 2008). In this section, the stemmers are evaluated using a manually stemmed dataset. In addition, an enhancement for the root extractors discussed in the Related Work section is obtained by using light stemmers as a preprocessing step to root extractors. The results of the enhancement are shown in tables 2, 3, and 4.

### 4.1 Evaluation Dataset

A set of 29 manually annotated documents were used for stemmers evaluation. The dataset is part of International Corpus of Arabic (ICA) (Alansary et al., 2007). ICA is a collection of Arabic documents obtained from various resources such as newspa-

pers, magazines, and books (Alansary et al., 2007). ICA was collected and annotated to give a complete representation of the Arabic language to be used in Arabic NLP research (Alansary et al., 2007). The 29 documents contain 10,302 tokens. Only 8,941 words are Arabic words, while the remaining are tokens, such as ”/T” (beginning of a title). Only 6,323 words have associated roots and 3,629 unique word-root pairs of the 10,302 tokens. Every word has a various set of features associated with it for evaluating the discussed stemmers, such as stem and root. This makes the dataset an ideal reference for evaluating the introduced stemmers. However, some features were left blank because the words do not have the associated feature, such as stopwords, as shown in Figure 1. The dataset will be used to conduct a series of comparisons to evaluate Arabic stemmers from various perspectives.

### 4.2 Evaluation Criteria

Arabic roots and stems provide a valuable set of characteristics that are useful for many computational tasks (Aljlayl and Frieder, 2002a)(Oraby et al., 2013a)(Ezzeldin et al., 2015). Various tasks require different perspectives such as Information Retrieval would use roots for grouping, and other may use linguistic features of roots.

The linguistic accuracy provides a representative measure for the efficiency of the stemmer in linguistic based tasks. Linguistic accuracy is computed as the ratio between the number of correctly stemmed words and the number of the input words. On the other hand, roots can be used as a word’s label, which can group linguistically similar words. Another set of measures were used to measure the macro and micro classification capabilities of the roots. The difference between macro and micro is that the size of class is reflected on the micro measure where the macro measure treats classes equally regardless of class’s size. The following set of equations are used to provide macro and micro classification measurements (Manning et al., 2008):

### 4.3 Evaluation Results

The stem and root features of the evaluation data set allow to investigate the two types of stemming algorithms, light stemming and root extraction. The light stemmers that are used in the experiment are Light10

Word	Lemmaid	Pr1	Pr2	Pr3	Stems	Tags	Suf1	Suf2	Root
21						Num			
الي	<ilaY				<ilaY	PREP			
27						Num			
الشهر	\$ahor	AI/DET			\$ahor	NOUN(ADV_1			\$hr
الحالي	HAliy~	AI/DET			HAliy~	ADJ			Hwl
.						Punc			
P/						EOF_Prg			
/P						BOF_Prg			
يفتح	{ifotataH	ya/IV3MS			fotatiH	IV	u/IVSUFF		ftH
فعليات	faE~Aliy~ap				faE~Aliy~	NOUN	At/NSUFF		fEl
المؤتمر	mu&otamar	AI/DET			mu&otamar	NOUN			'mr
السيد	say~id	AI/DET			say~id	NOUN			swd/syd
عمرو	Eamorw				Eamorw	NOUN_PROP			
موسى	muwsaY				muwsaY	NOUN_PROP			
الأمين	>amiyn	AI/DET			>amiyn	NOUN			'mn
العام	EAm~	AI/DET			EAm~	ADJ			Emm
لجامعة	jAmiEap	li/PREP			jAmiE	NOUN	ap/NSUFF		jmE
الدول	dawolap	AI/DET			duwal	NOUN			dwl
العربية	Earabiy~	AI/DET			Earabiy~	ADJ	ap/NSUFF		Erb
بمقر	maqar~	bi/PREP			maqar~	NOUN			qrr
الجامعة	jAmiEap	AI/DET			jAmiE	NOUN	ap/NSUFF		jmE
العربية	Earabiy~	AI/DET			Earabiy~	ADJ	ap/NSUFF		Erb
ويتم	tam~-i	wa/CONJ	ya/IV3MS		tim~	IV	u/IVSUFF		tmm
خلال	xilAl				xilAl	NOUN(ADV_1			xll
الافتتاح	{ifotitAH	AI/DET			{ifotitAH	NOUN			ftH
الإعلان	<iEolAn	AI/DET			<iEolAn	NOUN			Eln
عن	Ean				Ean	PREP			
الفائز	fA}iz	AI/DET			fA}iz	NOUN			fwz

Figure 1: Evaluation Dataset.

$$Accuracy_{macro} = \frac{\sum_{i=1}^n |X_i \cap Y_i|}{\sum_{i=1}^n |X_i \cup Y_i|} \quad Accuracy_{micro} = \frac{1}{n} \sum_{i=1}^n \frac{|X_i \cap Y_i|}{|X_i \cup Y_i|} \quad (1)$$

$$Precision_{macro} = \frac{\sum_{i=1}^n |X_i \cap Y_i|}{\sum_{i=1}^n |X_i|} \quad Precision_{micro} = \frac{1}{n} \sum_{i=1}^n \frac{|X_i \cap Y_i|}{|X_i|} \quad (2)$$

$$Recall_{macro} = \frac{\sum_{i=1}^n |X_i \cap Y_i|}{\sum_{i=1}^n |Y_i|} \quad Recall_{micro} = \frac{1}{n} \sum_{i=1}^n \frac{|X_i \cap Y_i|}{|Y_i|} \quad (3)$$

$$F_1_{macro} = \frac{\sum_{i=1}^n |X_i \cap Y_i|}{\sum_{i=1}^n (|X_i| + |Y_i|)} \quad F_1_{micro} = \frac{1}{n} \sum_{i=1}^n \frac{|X_i \cap Y_i|}{|X_i| + |Y_i|} \quad (4)$$

Where:

$n$  is the number of candidate roots.

$X$  is the set of candidate roots.

$X_i$  is an individual candidate root.

And

$Y$  is the set of (semantically) correct roots.

$Y_i$  is an individual (semantically) correct root.

Table 1: Light Stemmer Linguistic Accuracy

Light Stemmer	linguistic Accuracy
MADAMIRA (MADA)	91.73%
Light10	47.83%

Table 2: Stemmers Linguistic Accuracy

Stemmer	linguistic Accuracy	linguistic Coverage
Khoja	72.1%	72.1%
MADA + Khoja	72.1%	72.1%
ISRI	14.2%	14.2%
MADA + ISRI	16.91%	16.91%
Tashaphyne (TASH)	30.3%	30.3%
MADA + TASH	38.23%	38.23%
ElixirFM	NA	98.15%

Table 3: Stemmers Macro Classification Statistics

Stemmer	Accuracy	Precision	Recall	F <sub>1</sub>
Khoja	57.53%	57.53%	59.59%	58.55%
MADA + Khoja	57.53%	57.53%	59.59%	58.55%
ISRI	10.43%	10.43%	10.49%	10.46%
MADA + ISRI	15.40%	15.40%	15.60%	15.50%
TASH	25.07%	25.07%	25.15%	25.11%
MADA + TASH	41.79%	41.79%	41.85%	41.82%

Table 4: Stemmers Micro Classification Statistics

Stemmer	Accuracy	Precision	Recall	F <sub>1</sub>
Khoja	71.42%	95.38%	73.98%	83.33%
MADA + Khoja	71.42%	95.38%	73.98%	83.33%
ISRI	14.20%	97.34%	14.25%	24.87%
MADA + ISRI	17.25%	96.59%	17.36%	29.43%
TASH	30.42%	99.45%	30.47%	46.11%
MADA + TASH	39.61%	99.86%	39.62%	56.74%

and MADAMIRA stemmers, while the root extractors stemmers are Khoja, ISRI, and Tashaphyne. The experiments also investigate combining the two types of stemming algorithms, where light stemming is used as preprocessing for root extractors. It combines MADAMIRA stemmer with Khoja, ISRI, and Tashaphyne stemmers. Only unique words in the dataset having an associated root feature are used in the test.

Table 1 shows the improvement of the light stemming algorithm MADAMIRA over Light10 stemmer. MADAMIRA gives an accuracy of 91.73% with roughly 44% accuracy improvement over Light10. Using the MADAMIRA light stemmer as a pre-processing phase before root extraction using Khoja, ISRI, and Tashaphyne stemmers improves the accuracy of root extraction.

Table 2 shows the linguistic accuracy of Khoja,

ISRI, and Tashaphanye stemmers in standalone mode and when preceded by MADAMIRA pre-processing. There is a substantial difference between Khoja stemmer and the other two, ISRI and Tashaphyane, with at least 40% linguistic accuracy gap. This is due to the usage of a roots dictionary by Khoja. But when adding MADAMIRA as a pre-processing phase, there is a noticeable improvement in ISRI and Tashaphanye by roughly 2% and 8%, respectively. There is no effect of using MADAMIRA with Khoja, this is due to the robust segmentation of Khoja and the existence of dictionary validation. The ElixirFM morphological trees were not sufficient to disambiguate the generated roots. However, it provides a valuable set of features and substantial root converge which can be used for further analysis. The usage of MADAMIRA is also reflected on the classification and clustering measures. As noticed, the increase of linguistic accuracy increases related measures, namely, classification and clustering. Table 2 also shows the effectiveness of generating possible roots of ElixirFM. However, the syntactic strategy for distinguishing words' senses is not completely effective in producing only one root.

Classification has a distinctive property, that is grouping similar words. By comparing Tables 3 and 4, there is a noticeable increase in clustering measures over classification. This due to the fact that size of classes is being reflected to the micro classification measure where it is ignored with macro classification measure.

The performance of stemming algorithms can be noticeably improved by applying some minor changes, such as normalization processes. For example, changing the form of Hamaz (ء, an Arabic letter). Such changes would not affect only linguistic based task but also related non-linguistic tasks.

## 5 Conclusion

Stemmers are employed in various tasks, such as information retrieval (IR) (Aljlayl and Frieder, 2002a)(Larkey et al., 2002) and sentiment analysis (Oraby et al., 2013b). Stemmers achieve a noticeable improvement in related NLP tasks (Oraby et al., 2013a). However, the evaluation of stemmers does not explicitly show the stemming efficiency (Smirnov, 2008). In this paper, direct evaluation

was used to study the behaviour of Arabic stemmers. The paper investigates two types of stemming algorithms, namely, light stemmers and root extractors. The light stemmers studied were MADAMIRA (Pasha et al., 2014) and Light10 (Larkey et al., 2007). And, the root extractors studied were Khoja (Khoja and Garside, 1999), ISRI (Taghva et al., 2005), and Tashaphyane (Zerrouki, 2010). The measures used to compare the stemmers were the linguistic accuracy and coverage, in addition to macro and micro classification measures. The results obtained show that the increase of linguistic accuracy increases the effectiveness in other tasks(Oraby et al., 2013b)(Ezzeldin et al., 2015).

This study and IR's results (Taghva et al., 2005) show that low linguistic accuracy in stemming algorithms does not necessarily affect efficiency of a stemmer in information retrieval, possibly due to the presence frequently correct events (extracted roots). For example, ISRI stemmer has an accuracy of 14.2%, but performs efficiently and shows competitive result with Khoja in IR (Taghva et al., 2005). The study shows another set of possible improvements, which is using light stemmers as pre-processing for the root extraction task. Different studies show that light stemming has a higher potential for improving IR than root extraction (Larkey et al., 2002)(Taghva et al., 2005). Using light stemming associated with root extraction methods would build a complete hierarchical representation of Arabic words, in addition, light stemming improves the performance of other stemmers. The study conducted and the results obtained show the correlation between linguistic accuracy and other measures, the increase in linguistic accuracy increases other related measures. The existence of multiple Arabic stemmers adds richness to the stemming analysis task. Each of the discussed stemmers has its own strengths and weaknesses, where the weaknesses could be reduced by combining multiple stemmers in effective ways.

## References

- Sameh Alansary, Magdy Nagi, and Noha Adly. 2007. Building an international corpus of arabic (ica): progress of compilation stage.
- Mohammed Aljlayl and Ophir Frieder. 2002a. On arabic search: improving the retrieval effectiveness via a light stemming approach. pages 340–347
- Mohammed Aljlayl and Ophir Frieder. 2002b. On arabic search: improving the retrieval effectiveness via a light stemming approach. pages 340–347
- Imad A AlSughaiyer and Ibrahim A AlKharashi. 2004. Arabic morphological analysis techniques: A comprehensive survey. *Journal of the American Society for Information Science and Technology*, 55(3):189–213
- Tim Buckwalter. 2002. Buckwalter Arabic morphological analyzer version 1.0.
- Kareem Darwish. 2002. Building a shallow arabic morphological analyzer in one day. pages 1–8.
- Mona Diab, Kadri Hacioglu, and Daniel Jurafsky. 2007. Automated methods for processing arabic text: from tokenization to base phrase chunking. *Arabic Computational Morphology: Knowledge-based and Empirical Methods*. Kluwer/Springer.
- Mahmoud El-Defrawy, Yasser El-Sonbaty, and Nahla A Belal. 2015. Cbas: Context based arabic stemmer. *International Journal on Natural Language Computing (IJNLC)*.
- Ahmed Magdy Ezzeldin, Mohamed Hamed Kholief, and Yasser El-Sonbaty. 2013. Alqasim: Arabic language question answer selection in machines. In *Information Access Evaluation. Multilinguality, Multimodality, and Visualization*, volume 8138, pages 100–103 Springer.
- Ahmed Magdy Ezzeldin, Yasser El-Sonbaty, and Mohamed Hamed Kholief. 2015. Exploring the effects of root expansion, sentence splitting and ontology on arabic answer selection. *Natural Language Processing and Cognitive Science: Proceedings 2014*, page 273
- Radwa Fathalla, Yasser El-Sonbaty, and Mohamed A Ismail. 2007. Extraction of arabic words from complex color image. In *9th IEEE International Conference on Document Analysis and Recognition (ICDAR 2007)*, pages 1223–1227, Brazil, 23–26 September. IEEE.
- Nizar Habash, Owen Rambow, and Ryan Roth. 2009. Mada+ token: A toolkit for arabic tokenization, diacritization, morphological disambiguation, pos tagging, stemming and lemmatization. In *Proceedings of the 2nd International Conference on Arabic Language Resources and Tools (MEDAR)*, Cairo, Egypt, pages 102–109.
- John Hutchins. 2004. The first public demonstration of machine translation: the georgetown-ibm system, 7th january 1954.
- Hongyan Jing and Kathleen R McKeown. 2000. Cut and paste based text summarization. pages 178–185.
- Shereen Khoja and Roger Garside. 1999. Stemming arabic text. *Lancaster, UK, Computing Department, Lancaster University*.
- Leah S Larkey, Lisa Ballesteros, and Margaret E Connell. 2002. Improving stemming for arabic information retrieval: light stemming and co-occurrence analysis. pages 275–282
- Leah S Larkey, Lisa Ballesteros, and Margaret E Connell. 2007. Light stemming for arabic information retrieval. In *Arabic computational morphology*, pages 221–243 1402060459. Springer.
- Christopher D Manning, Prabhakar Raghavan, and Hinrich Schütze. 2008. *Introduction to information retrieval*, volume 1. Cambridge university press Cambridge.
- Will Monroe, Spence Green, and Christopher D Manning. 2014. Word segmentation of informal arabic with domain adaptation. *ACL, Short Papers*.
- Prakash M Nadkarni, Lucila Ohno-Machado, and Wendy W Chapman. 2011. Natural language processing: an introduction. *Journal of the American Medical Informatics Association*, 18(5):544–551
- Ani Nenkova. 2005. Automatic text summarization of newswire: Lessons learned from the document understanding conference. 5:1436–1441.
- Shereen Oraby, Yasser El-Sonbaty, and Mohamad Abou El-Nasr. 2013a. Finding opinion strength using rule-based parsing for arabic sentiment analysis. In *Advances in Soft Computing and Its Applications*, volume 8266, pages 509–520
- Shereen M Oraby, Yasser El-Sonbaty, and Mohamad Abou El-Nasr. 2013b. Exploring the effects of word roots for arabic sentiment analysis. In *Conference on Natural Language Processing*, Nagoya, Japan, October.
- Arfath Pasha, Mohamed Al-Badrashiny, Ahmed El Kholy, Ramy Eskander, Mona Diab, Nizar Habash, Manoj Pooleery, Owen Rambow, and Ryan Roth. 2014. Madamira: A fast, comprehensive tool for morphological analysis and disambiguation of arabic. In *In Proceedings of the 9th International Conference on Language Resources and Evaluation, Reykjavik, Iceland*.
- Karin C 2005. *A reference grammar of modern standard Arabic*. Cambridge University Press.
- Sherine Nagi Saleh and Yasser El-Sonbaty. 2007. A feature selection algorithm with redundancy reduction for text classification. In *Computer and information sciences, 2007. iscis 2007. 22nd international symposium on*, pages 1–6
- Iliia Smirnov. 2008. Overview of stemming algorithms. *Mechanical Translation*.

- Otakar Smrz, Viktor Bielický, Iveta Kourilová, Jakub Krácmár, Jan Hajic, and Petr Zemánek. 2008. Prague arabic dependency treebank: A word on the million words. In *Proceedings of the Workshop on Arabic and Local Languages (LREC 2008), Marrakech, Morocco*, pages 16–23.
- Otakar Smrž. 2007. Elixirfm: implementation of functional arabic morphology. In *Proceedings of the 2007 Workshop on Computational Approaches to Semitic Languages: Common Issues and Resources*, pages 1–8. Association for Computational Linguistics.
- Kazem Taghva, Rania Elkhoury, and Jeffrey S Coombs. 2005. Arabic stemming without a root dictionary. pages 152–157.
- Taha Zerrouki. 2010. Tashaphyne, arabic light stemmer/segment. <http://tashaphyne.sourceforge.net>.

## Where Morphological Complexity Matters

**Tomokazu Takehisa**

Niigata University of Pharmacy and Applied Life Sciences

265-1 Higashijima, Akiha-ku, Niigata 956-8603, Japan

takehisa@nupals.ac.jp

### Abstract

It has been long observed that Latinate verbs in English cannot appear in verb-particle constructions, resultative constructions, and double object constructions. Recent research has revealed that, despite persistent counterexamples, the hypothesis invoking the morphological complexity of verbs is the most promising in dealing with the Latinate/native asymmetry (Coppock, 2009; Harley, 2008; Punske 2012, 2013). This paper aims to show two more cases of the asymmetry in favor of the morphological complexity hypothesis. Moreover, in an attempt to refine the hypothesis, an analysis will be provided within Distributed Morphology (Halle and Marantz, 1993). Specifically, it argues that the asymmetry can be reduced to the selectional properties of the acategorical roots involved. Some roots are obligatorily specified for a particular morpheme and combine with it to form a complex root, while others are not necessarily specified as such and they can either stand alone as a simple root or form a complex root.

### 1. Introduction

It has been long observed that Latinate verbs in English are typically bad with verb-particle constructions (e.g., Whorf, 1956; Di Sciullo and Williams, 1987; Harley, 2008), resultative constructions (Harley, 2008), and double object constructions (Pinker, 1989; Pesetsky, 1995; Harley, 2008), which are commonly found in Germanic languages and considered to be a family

of constructions (Snyder, 1995; Stromswold and Snyder, 1995; Snyder and Stromswold, 1997).

Various hypotheses have been proposed to derive the asymmetry observed between Latinate and native verbs in those constructions. Notable among them are the prosodic weight hypothesis, which takes the prosodic weight of verbs as a crucial factor (Grimshaw, 2005; Anttila, 2007), the two-lexicon hypothesis, which makes recourse to two different lexical classes, Latinate and native (Grimshaw, 2005),<sup>1</sup> and the morphological complexity hypothesis, which invokes the morphological complexity of verbs (Pinker, 1989; Harley, 2008).<sup>2</sup> While there are persistent counterexamples, the morphological complexity hypothesis, as it stands, is the most promising hypothesis in dealing with the Latinate/native asymmetry, as convincingly demonstrated in a series of psycholinguistic experiments on ditransitivity by Coppock (2009).

In this paper, assuming that the morphological complexity hypothesis is on the right track, I will attempt to further increase the plausibility of the hypothesis. Specifically, I will show that the Latinate/native asymmetry can be observed in two more empirical domains, along with the constructions mentioned above: exocentric V-N compounds and non-compositional verb phrase idioms. Moreover, an analysis will be presented within the framework of Distributed Morphology (henceforth, DM; Halle and Marantz, 1993). Specifically, I will argue that the difference between Latinate and native verbs can be reduced

<sup>1</sup> Grimshaw uses the terms Romance and Germanic for Latinate and native, respectively.

<sup>2</sup> Coppock's (2009) classification of the hypotheses is adopted.

to the selectional properties of the acategorical roots involved: some roots are obligatorily specified for a particular morpheme and form a complex root, while others are not necessarily specified as such and they can stand alone as a simple root or form a complex root, depending on the specification.

The organization of the paper is as follows: In the next section, I will briefly review the observed asymmetry between Latinate and native verbs in English. In section 3, I will demonstrate that the same asymmetry can be observed in two more empirical domains, exocentric V-N compounds and non-compositional verb phrase idioms. Moreover, an analysis will be provided to account for the Latinate/native asymmetry in terms of the selectional properties of the roots involved. Section 4 is a summary.

## 2. The Latinate/Native Asymmetry

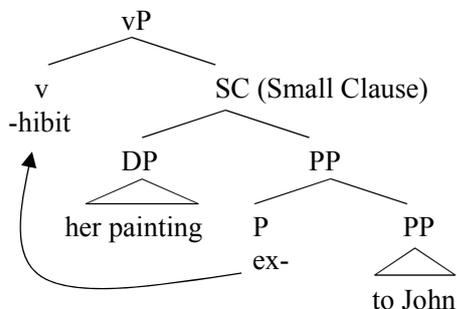
### 2.1 Verb Particle Constructions

As mentioned above, Latinate verbs cannot form a verb-particle combination in general, as shown by the following examples, taken from Harley (2008).

- (1) a. write it up      \*compose/arrange it up
- b. eat it up        \*consume it up
- c. finish it up     \*complete it up
- d. throw it out    \*discard it out
- e. show it off      \*exhibit it off

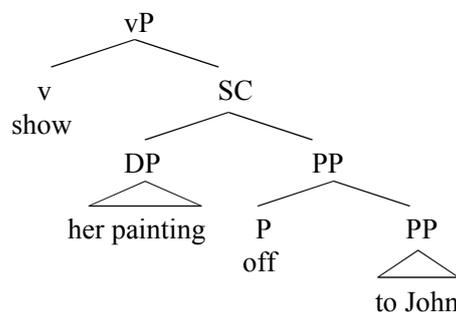
The robust contrast in (1) strongly suggests that Latinate and native verbs are significantly different at some level. To account for their difference, Harley (2008) proposes the following structures for Latinate verbs and particle verbs, respectively:<sup>3</sup>

- (2) Mary exhibited her painting to John.



<sup>3</sup> Harley (2008) assumes that acategorical roots, e.g.  $\sqrt{-hibit}$  and  $\sqrt{show}$ , are inserted into v (Manner Incorporation).

- (3) Mary showed off her painting to John.



For Harley, the two kinds of verbs are structurally the same, with the assumption that a Latinate prefix, *ex-* in (2), and a particle, *off* in (3), are the same, with the only difference being that the former incorporates into the verb, and the restriction on Latinate verbs can be explained in terms of structural competition: the particle and the prefix cannot co-occur because they occupy the same structural position. Thus, in one sense, this analysis directly implements Cowie and Mackin's (1979) pre-theoretical intuition that particle verbs and Latinate affixed verbs are on a par.

Straightforward as it may be, there are a number of Latinate verbs which run counter to the morphological complexity hypothesis. Consider (4) and (5).<sup>4</sup>

- (4) a. centrifuge it out
- b. partition it off
- c. telegraph back/in
- d. telephone around/back/in/over
- (5) a. divide it up
- b. collect it up
- c. conduct them away
- d. entice them away
- e. separate them out

(Shimada, 1985)

The verbs in (4) are instrumental denominal verbs and they may well receive a different analysis from the one in (2). Specifically, while they are

<sup>4</sup> It should be made clear that monomorphemic Latinate verbs can form (idiomatic) verb-particle combinations, as in (i).

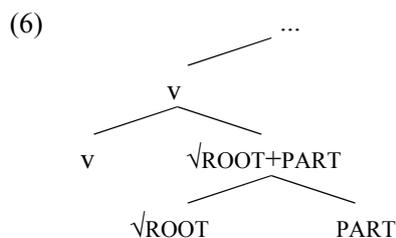
- (i) a. seize up
- b. serve it up, serve it out
- c. figure it out

This clearly shows that morphological complexity matters, not the etymological origin.

morphological complex, denominal verbs are immune to stress assignment in the verbal domain and retain their nominal stress pattern, which suggests that they are not in the domain where their morphological complexity matters.<sup>5</sup>

On the other hand, it appears the verbs in (5) do run counter to the morphological complexity hypothesis. Although I do not have a satisfactory answer at present as to how to accommodate them under the hypothesis, one thing worth pointing out is that the combinations in (5) (and those in (4)) do not have idiomatic interpretations, which are typically observed with verb-particle combinations, and the particles involved retain their meaning, either directional or aspectual (Jackendoff, 2002).

In an attempt to make (5) make look less strange, I propose that, in addition to (2) and (3), the following structure is also possible for Latinate verbs and particle verbs.<sup>6, 7</sup>



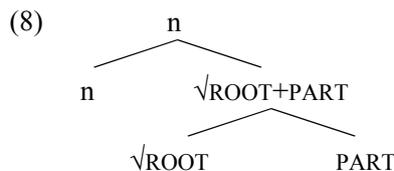
The structure in (6), where a root and a (prefixed) particle forms a complex predicate,<sup>8</sup> has its basis on the consensus in the literature that verb-particle combinations are associated with two different configurations, which are reflected in their interpretations (Wurmbrand, 2000; Basilico, 2008, among others). Specifically, the compositional interpretation and the idiomatic/non-compositional interpretation are associated with the structure in (3) and the Germanic version of (6), respectively. Thus, I extend this distinction into Latinate verbs by assuming they can be associated with the structures in (2) and (6). This assumption makes it possible to have two particles, one idiomatic and the other compositional in the verbal domain,

thereby opening up the possibility for accounting for the combinations in (5).

Independent support for (6) as a possible structure for particle verbs comes from nominalized verb-particles, or exocentric V-P compounds, as in (7) below. It is highly unlikely to derive them by syntactic incorporation, which is assumed in (2) (Farrell, 2005).

- (7) a. a drop off
- b. a show off
- c. a break up
- d. a hold up
- e. a set back

Moreover, within DM, it is assumed that acategorical roots have their categorial status determined by category-defining functional heads such as *v*, *a* and *n* (Marantz, 2001), and the nouns in (7) are analyzed to have the following structure, parallel to their verbal counterparts.



According to Marantz (2001), those category-defining heads fix the edge of a cyclic domain whereby the interpretation of the root in the context of the categorizing functional head is negotiated, using the encyclopedic knowledge. This view of interpretation is quite congenial to the proposal in the verb-particle literature that idiomatic verb-particle combinations involve a complex head structure, as in (6). We will turn to this point below.

### 2.2 Resultative Constructions

The Latinate/native asymmetry can also be observed in the case of resultative constructions. Consider the following examples from Harley (2008).

- (9) a. fill it full                    \*inflate it full
- b. squeeze it empty        \*compress it empty
- c. stab it dead               \*impale it dead
- d. eat yourself sick        \*devour yourself sick
- e. freeze solid               \*congeal solid

<sup>5</sup> See Kiparsky (1982, 1997) and Arad (2005).  
<sup>6</sup> I remain agnostic about Harley’s treatment of Manner Incorporation, simply treating category-defining functional heads and acategorical roots as separate terminal nodes.  
<sup>7</sup> I assume, following Zhang (2007) and Basilico (2008), that a root can be complex before the categorizing head is merged.  
<sup>8</sup> The PART head gets realized as a prefix after syntax.

By assuming the structure as in (3), with a resultative predicate as the predicate of a small clause in place of a particle, Harley successfully accounts for the asymmetry in terms of structural competition.

It is worth mentioning in this connection that some resultatives can behave more like verb-particles in that a native verb and a resultative adjective form a complex predicate, without the help of Heavy NP Shift, as shown in (10).<sup>9</sup> In such cases, it is appropriate to treat them as involving the structure in (6).

- (10) a. John cuts open the melon.
- b. The activists set free the lab rats.
- c. John wiped clean the table.

### 2.3 Double Object Constructions

Pesetsky (1995) made the observation that Latinate verbs are awkward with double objects, as shown in (11):

- (11) a. Susie gave Oxfam some canned food.
- a'. Susie gave some canned food to Oxfam.
- b.\*Susie donated Oxfam some canned food.
- b'. Susie donated some canned food to Oxfam.
- c. Bill sent Sue his regards.
- c'. Bill sent his regards to Sue.
- d.\*Bill conveyed Sue his regards.
- d'. Bill conveyed his regards to Sue.
- e. Mary showed the committee her findings.
- e'. Mary showed her findings to the committee.
- f.\*Mary displayed the committee her findings.
- f'. Mary displayed her findings to the committee.
- g. Tom told Ben the story.
- g'. Tom told the story to Ben.
- h.\*Tom recounted Ben the story.
- h'. Tom recounted the story to Ben.

(Pesetsky, 1995: 128ff.)

As is the case with verb-particles and resultatives, Harley accounts for the asymmetry in terms of structural competition by assuming that low Appl(icative)P (Pylkkänen, 2008) is involved in place of SC in (2).

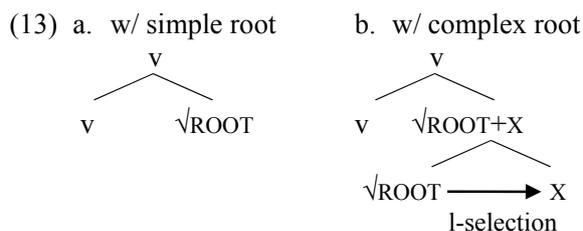
<sup>9</sup> The examples are from Neeleman (1992), Svenonius (1994), and Williams (1997), respectively.

There are a number of attested counterexamples to the asymmetry. A partial list of Latinate verbs entering into the double object construction is given in (12), taken from Harley (2008). I have nothing insightful to say about these verbs at present, only hoping to accommodate them under the morphological complexity hypothesis.

- (12) allot, assign, bequeath, concede, extend, reduce, etc.

### 3. More on the Asymmetry

So far, we have considered the observations made in the literature pertaining to the Latinate/native asymmetry. In the three cases we saw, the asymmetry is attributed to the difference in the structure of the root domain between Latinate and native verbs. By and large, Latinate verbs involve complex roots, while native verbs simple roots. This difference is ultimately reduced to the selectional properties of Latinate roots. Specifically, the type of selection that is relevant is l(exical)-selection, i.e., selection for particular lexical items (Pesetsky, 1995; Everaert, 2010).<sup>10</sup> Thus, particular Latinate roots are obligatorily specified for a set of particular morphemes and combine with them to form complex roots, as in (13)b; native roots can also be specified as such, as in the case of idiomatic verb-particle constructions, but they can be unspecified and stand alone as a simple root, as in (13)a, as well. These roots undergo categorization by v, also shown in (13).



Once we accept the structures in (13), we can immediately explain the fact that Latinate verbs are

<sup>10</sup> L-selection is defined in Everaert (2010:94) as follows:  
 (i) a. L-selection involves the selection by one terminal element  $\alpha$  of another terminal element  $\beta$  where the projection of  $\beta$  is in the syntactic domain of  $\alpha$ .  
 b. The syntactic domain of head  $\alpha$  is the set of nodes contained in  $\text{Max}(\alpha)$  that are distinct from and do not contain  $\alpha$ .

more restricted in distribution than native verbs. Thus, simple roots, as in (13)a, can combine with some morpheme to form a complex root, as in (13)b, but the latter cannot form a (more) complex root because they are already complex.<sup>11</sup> In this sense, the structure in (13)b reflects the view that verb-particle combinations and prefixed Latinate verbs are on a par.

This said, we will see in this section two more cases of the Latinate/native asymmetry and that the above difference in structure plays a crucial role in deriving the asymmetry.

### 3.1 Exocentric V-N Compounds

Recall that in section 2.1, we discussed exocentric V-P compounds and that they are associated with idiosyncratic interpretations. In what follows, I will argue that essentially the same analysis applies to exocentric V-N compounds, with twists on the interpretation.

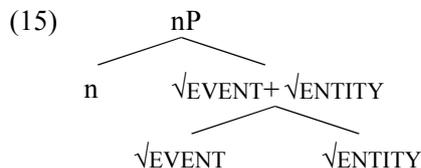
First, let us see the relevant data. Consider (14).

- (14) a. breakwater      \*disintegrate-water
- b. cutthroat        \*incise-throat
- c. killjoy          \*eliminate-joy
- d. kiss-ass        \*osculate-ass
- e. turnkey         \*rotate-key
- f. scarecrow      \*intimidate-crow

While it is impossible to argue for the absence of something, the examples in (14) give you the feel that Latinate verbs cannot form V-N compounds.

Thus, I claim that the gap is not accidental and Latinate verbs are systematically ruled out as the V part of a V-N compound.<sup>12</sup> Moreover, given the assumption on categorization in DM, I argue that V-N compounds do not involve categorized morphemes, [*v* √ROOT] or [*n* √ROOT], but a root naming an event and another root naming an entity.<sup>13,14</sup> Viewed this way, an exocentric V-N

compound is a nominalization with a complex root, as in (15):



This straightforwardly explains why Latinate verbs cannot form V-N compounds: a root naming an event must be a simple root, as in (13)a, not a complex one, as in (13)b.

Given the observation that the N of a V-N compound is construed as the object of the V, one might wonder how the interpretation can be derived from the structure in (15). I claim the observed interpretation is not obtained from the rigid argument structure of the V, but through negotiation with the encyclopedic knowledge relating to the event root involved at the conceptual interface (Marantz, 2001; Barker, 1998). Moreover, it has also been observed the compound noun itself is construed as agentive or instrumental, denoting a human, an animal or a thing. I assume that the nominalizing head involved in the compound denotes an entity which is construed as participating in an event of the type related to the episodic content of the complex root. Given this, since the complex root involved roughly corresponds to an event which involves a proto-patient, the denotation of the compound noun can only be construed as participating in the event as a proto-agent, i.e., as agentive or instrumental. Thus, we can provide an explanation for the interpretation of V-N compounds without making recourse to the argument structure of the V.

Although our primary focus is on English in this paper, what we have argued for in the case of exocentric V-N compounds seems to hold in other languages as well. For instance, in French, a complex root cannot appear as the V part of a V-N compound. Consider the following:<sup>15</sup>

- (16) a. grille-pain                      b. ouvre-boîte
- grill-bread                    open-can
- 'toaster'                        'can opener'

<sup>11</sup> No recursive property is observed in this domain. This indicates that the root domain is relevant to the phenomena.

<sup>12</sup> Here, terms such as “V” and “N” are used as descriptive cover terms, not theoretical entities.

<sup>13</sup> I follow Harley (2005) and Anagnostopoulou and Samioti (2014) that roots can be classified into three ontological types: events, states, and entities.

<sup>14</sup> It is possible to assume that the nominal in a compound is categorized as a noun by *n*, if one adopts a layered structure for nominal complex DP hypothesis where DP contains Num(ber)P, which contains NP, here nP.

<sup>15</sup> The glosses are simplified in such a way that theme vowels, which are at the end of the verb stem, are treated as part of the verb stem.

- |                  |                       |                    |
|------------------|-----------------------|--------------------|
| c. tourne.vis    | d. leche-vitrine      | e. spill the beans |
| turn.screw       | lick-window           | f. pull strings    |
| ‘screwdriver’    | ‘window shopping’     | g. break the ice   |
| e. gratte-papier | f. coupe-gorge        | h. kick the habit  |
| scratch-paper    | cut-throat            |                    |
| ‘pen pusher’     | ‘dangerous back ally’ |                    |

- (17) a. **remue-menage**  
 move-housework  
 ‘commotion, bustle’  
 b. **reveille-matin**  
 wake.up-morning  
 ‘alarm clock’

Although the examples in (17) run counter to the generalization and need to be accommodated in some way or other under the morphological complexity hypothesis, it largely holds true in French as well that a complex root cannot appear as the V part of a V-N compound.

The evidence from exocentric V-N compounds is significant in that they are not particular to Germanic languages, unlike the cases we have seen so far, i.e., resultatives, verb-particles, and double object constructions, clearly showing that the asymmetry can be observed with non-Germanic languages as well. Thus, although we have referred to the observed asymmetry as the Latinate/native asymmetry, it is ultimately not about the vocabulary type or etymological origin, but rather about the complexity of the verb/event root involved, as depicted in (13) above.

### 3.2 Verb Phrase Idioms

The other hitherto unnoticed case of Latinate/native asymmetry involves verb phrase idioms (henceforth, VP idioms). It goes without saying that English has countless VP idioms, but, when you examine them more closely, you notice that you seldom find idioms based on Latinate verbs. This may sound trivial due to the fact that many idioms involve verb-particle combinations, which are mostly incompatible with Latinate verbs, as we have seen above. Yet the observation holds in other cases which involve a verb and its object as well. First, consider (18):

- (18) a. kick the bucket  
 b. bite the dust  
 c. carry the can  
 d. jump the shark

The native verb idioms in (18) can be classified into two classes: non-compositional idioms, as in (18)a–(18)d, and compositional idioms, as in (18)e–(18)h. Compositional idioms are said to have meanings distributed among their parts and the correspondences between literal and idiomatic meanings can be made, while no such correspondences hold in non-compositional idioms and the idiomatic expression as a whole is associated with a particular idiosyncratic meaning (Nunberg et al., 1994).

VP idioms with Latinate verbs are hard to find, but the following examples can be considered to be idioms in terms of conventionality.

- (19) a. deliver the goods  
 b. connect the dots  
 c. contemplate one’s navel  
 d. deserve a medal (for doing)  
 e. reinvent the wheel  
 f. recharge your batteries  
 g. promise someone the moon

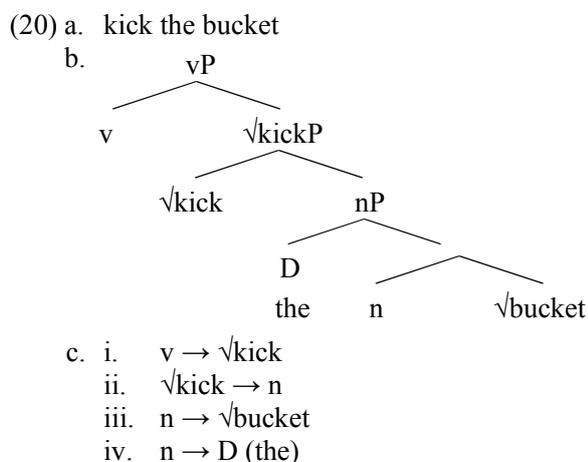
However, these examples are highly compositional, and the literal meanings of the parts of these idioms play an important role in interpretation in that they are mapped to the idiomatic meanings. To put it differently, in (19), the idiomatic meanings cannot be obtained without accessing the literal or original meanings.

Thus, the asymmetry in VP idioms can be stated as follows: idioms with native verbs can be compositional or non-compositional, while those with Latinate verbs can only be compositional.

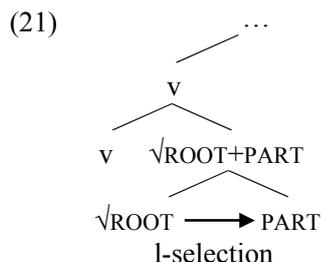
Since there are not so many non-compositional VP idioms to begin with, the gap may be accidental. Yet a paucity of compositional idioms is also a telling piece of evidence that shows that Latinate verbs adhere to their literal meanings, which are in some sense idiomatic to being with.

This asymmetry can also be captured in terms of the structural difference between simple and complex roots in (13). In fact, a coherent picture can be drawn of the incompatibility of non-compositional VP idioms with Latinate verbs by invoking the l-selectional properties of the

morphemes involved. Specifically, contemporary accounts of idioms hold that the structural constraints on idioms can be reduced to the l-selectional properties of particular morphemes (O’Grady, 1998; Everaert, 2010, Bruening, 2010), and also that a chain of selectional relations, which are structurally established by means of head-to-head relations, provides the basis for special, idiomatic meaning (O’Grady, 1998). Thus, an idiom like *kick the bucket* under the current assumptions has the following chain of relations, as given in (20)c, based on the structure in (20)b.



Turning to Latinate verbs with complex roots, as in (13)b, repeated here as (21), we can see why Latinate verbs do not form non-compositional idioms: they are idioms in their own right, with no further material involved. Specifically, an event root l-selects a particular prefix, represented as PART in (21), which l-selects nothing. As we have seen above, the complex root, or a chain of l-selection in this case, is negotiated in the context of v by using the encyclopedic knowledge in order to derive special meaning. This analysis can be regarded as a contemporary rendition of Katz and Postal’s (1963) intuition pertaining to what they call lexical idioms.



Note that the compositional idioms as in (19) have the original meanings, read off from the structure in (21), mapped into their idiomatic meanings in the context of some other particular morphemes.

VP idioms are not particular to Germanic languages. Once again, examples from French are given in (22) and (23) below. Note that the examples in (23) do not run counter to the morphological complexity hypothesis, as long as they are compositional idioms.

- (22) a. casser sa pipe  
 break one’s pipe  
 ‘kick the bucket, die’  
 b. griller un feu rouge  
 grill a fire red  
 ‘run a red light’  
 c. lever le coude  
 raise the elbow  
 ‘enjoy a drink’
- (23) a. promettre la lune  
 promise the moon  
 ‘promise the moon’  
 b. retourner sa veste  
 return one’s jacket  
 ‘become a turncoat’

#### 4. Summary

Drawing heavily on Harley (2008), we have attempted in this paper to further increase the plausibility of the morphological complexity hypothesis. Specifically, I have shown that exocentric V-N compounds and non-compositional VP idioms display the Latinate/native asymmetry and accounted for the asymmetry in terms of the structural difference between simple and complex roots, as given in (13). I have also argued that the difference in (13) should be ultimately attributed to the l-selectional properties of the roots involved. Specifically, although some roots are specified for particular morphemes and form complex roots, others have no intrinsic l-selectional properties and hence stand alone as simple roots. Such simple roots are commonly found in English and other Germanic languages, and they can enter into complex predicate formations such as resultatives, verb-particles, and double object constructions. Moreover, as we have shown above, simple roots

can form exocentric V-N compounds and non-compositional VP idioms, both of which can be in a way regarded as “complex root formation.” On the other hand, roots with their intrinsic 1-selectional properties obligatorily form complex roots by combining with particular morphemes, e.g., prefixes in Romance languages. Such roots, as they can be complex only in the specified ways, cannot enter into the aforementioned formations freely.

As we have seen above, there are a number of attested counterexamples to the morphological complexity hypothesis. Hopefully, further inquiry into such examples will lead to refinement—rather than confutation—of the hypothesis.

### Acknowledgments

I am grateful to Chigusa Morita and the three anonymous reviewers for invaluable comments and suggestions on an earlier version of this paper. I am solely responsible for the inadequacies that remain.

### References

- Anagnostopoulou, Elena and Yota Samioti. 2014. Domains within Words and Their Meanings: A Case Study. In A. Alexiadou, H. Borer and F. Schäfer, eds., *The Syntax of Roots and the Roots of Syntax*, pp.81-111. Oxford University Press, Oxford.
- Anttila, Arto. 2008. Phonological Constraints on Constituent Ordering. In C.B. Chang and H.J. Haynie, eds., *Proceedings of the 26th West Coast Conference on Formal Linguistics*, pp.51-59. Cascadilla Proceedings Project, Somerville, MA.
- Arad, Maya. 2005. *Roots and Patterns: Hebrew Morpho-syntax*. Springer, Dordrecht.
- Barker, Chris. 1998. Episodic *-ee* in English: A Thematic Role Constraint on New Word Formation. *Language*, 74(4):695-727.
- Basilico, David. 2008. Particle Verbs and Benefactive Double Objects in English: High and Low Attachments. *Natural Language and Linguistic Theory*, 26(4):731-773.
- Bruening, Benjamin. 2010. Ditransitive Asymmetries and a Theory of Idiom Formation. *Linguistic Inquiry*, 41(4):519-562.
- Coppock, Elizabeth. 2009. The Logical and Empirical Foundations of Baker’s Paradox. Ph.D. Thesis, Stanford University.
- Cowie, Anthony P. and Ronald Mackin. 1979. *Oxford Dictionary of Current Idiomatic English, Volume 1: Verbs with Prepositions and Particles*. Oxford University Press, Oxford.
- Di Sciullo, Anna Maria, and Edwin Williams. 1987. *On the Definition of Word*. MIT Press, Cambridge, MA.
- Everaert, Martin. 2010. The Lexical Encoding of Idioms. In M. Rapaport Hovav, E. Doron, and I. Sichel, eds., *Lexical Semantics, Syntax, and Event Structure*, pp.76-98. Oxford University Press, Oxford.
- Farrell, Patrick. 2005. English Verb-Preposition Constructions: Constituency and Order. *Language* 81(1):96-137.
- Grimshaw, Jane. 2005. *Words and Structure*. CSLI Publications, Stanford, CA
- Harley, Heidi. 2005. How Do Verbs Take Their Names? Denominal Verbs, Manner Incorporation and the Ontology of Roots in English. In N. Erteschik-Shir and T. Rapoport, eds., *The Syntax of Aspect*, pp.42-62. Oxford University Press, Oxford.
- Harley, Heidi. 2008. The Bipartite Structure of Verbs Cross-linguistically, Or, Why Mary Can’t Exhibit John Her Paintings. In T. Cristófaró Silva and H. Mello, eds., *Conferências do V Congresso Internacional da Associação Brasileira de Linguística*, pp.45-84. ABRALIN and FALÉ/UFMG, Belo Horizonte, Brazil.
- Halle, Morris and Alec Marantz. 1993. Distributed Morphology and the Pieces of Inflection. In K. Hale and S.J. Keyser, eds., *The View from Building 20*, pp.111-176. MIT Press, Cambridge, MA.
- Jackendoff, Ray. 2002. English Particle Constructions, the Lexicon, and the Autonomy of Syntax. In N. Dehé, R. Jackendoff, A. McIntyre, and S. Urban, eds., *Verb-Particle Explorations*, pp.67-94. Mouton de Gruyter, Berlin/New York.
- Katz, Jerrold J. and Paul M. Postal. 1963. Semantic Interpretation of Idioms and Sentences Containing Them. *Quarterly Progress Report of the Research Laboratory of Electronics*, 70, pp.275-282. MIT.
- Kiparsky, Paul. 1982. Word Formation and the Lexicon. In F. Ingeman, ed., *Proceedings of the Mid-America Linguistics Conference*, pp.3-29. University of Kansas.
- Kiparsky, Paul. 1997. Remarks on Denominal Verbs. In A. Alsina, J. Bresnan and P. Sells, eds., *Argument Structure*, pp.473-499. CSLI Publications, Stanford.
- Marantz, Alec. 2001. Words. Paper presented at the 20<sup>th</sup> West Coast Conference on Formal Linguistics, University of Southern California, 23-25, February.

- Neeleman, Ad. 1992. Complex Predicates. Ph.D. Thesis, Utrecht University.
- Nunberg, Geoffrey, Ivan A. Sag, and Thomas Wasow. 1994. Idioms. *Language*, 70(3):491-538.
- O'Grady, William. 1998. The Syntax of Idioms. *Natural Language and Linguistic Theory*, 16(2):279-312.
- Pesetsky, David. 1995. *Zero Syntax*. MIT Press, Cambridge, MA.
- Pinker, Stephen. 1989. *Learnability and Cognition: The Acquisition of Argument Structure*. MIT Press, Cambridge, MA.
- Punske, Jeffrey. 2012. Aspects of the Internal Structure of Nominalization: Roots, Morphology and Derivation. Ph.D. Thesis, University of Arizona.
- Punske, Jeffrey. 2013. Three Forms of English Verb Particle Constructions. *Lingua* 135:155-170.
- Pylkkänen, Liina. 2008. *Introducing Arguments*. MIT Press, Cambridge, MA.
- Shimada, Hiroshi. 1985. *Kudoosi (Phrasal Verbs)*. Taishukan, Tokyo.
- Snyder, William. 1995. A Neo-Davidsonian Approach to Resultatives, Particles, and Datives. In J. Beckman, ed., *Proceedings of NELS 25*, pp.457-472. GLSA, Amherst, MA.
- Snyder, William, and Karin Stromswold. 1997. The Structure and Acquisition of English Dative Constructions. *Linguistic Inquiry*, 28(2):281-317.
- Stromswold, Karin, and William Snyder. 1995. Acquisition of Datives, Particles, and Related Constructions: Evidence for a Parametric Account. In D. MacLaughlin, and S. McEwen, eds., *Proceedings of the 19<sup>th</sup> Annual Boston University Conference on Language Development*, pp.621-628. Cascadilla Press, Somerville, MA.
- Svenonius, Peter. 1994. Dependent Nexus. Ph.D. Thesis, University of California, Santa Cruz.
- Whorf, Benjamin. 1956. *Language, Thought, and Reality*. MIT Press, Cambridge, MA.
- Wurmbrand, Susi. 2000. The Structure(s) of Particle Verbs. Ms. McGill University, Montréal.
- Zhang, Niina Ning. 2007. Root Merger in Chinese Compounds. *Studia Linguistica*, 61(2):170-184.

## Distinguishing between True and False Stories using various Linguistic Features

**Yaakov HaCohen-Kerner**

Dept. of Computer Science  
Jerusalem College of Technology  
21 Havaad Haleumi St., P.O.B. 16031  
9116001 Jerusalem, Israel  
kerner@jct.ac.il

**Rakefet Dilmon**

Dept. of Hebrew  
and Semitic Languages  
Bar-Ilan University  
5290002 Ramat-Gan, Israel  
rak2@bezeqint.net

**Shimon Friedlich**

Dept. of Computer Science  
Jerusalem College of Technology  
21 Havaad Haleumi St., P.O.B. 16031  
9116001 Jerusalem, Israel  
shimonfriedlich@gmail.com

**Daniel Nissim Cohen**

Dept. of Computer Science  
Jerusalem College of Technology  
21 Havaad Haleumi St., P.O.B. 16031  
9116001 Jerusalem, Israel  
sdanielco@gmail.com

### Abstract

This paper analyzes what linguistic features differentiate true and false stories written in Hebrew. To do so, we have defined four feature sets containing 145 features: POS-tags, quantitative, repetition, and special expressions. The examined corpus contains stories that were composed by 48 native Hebrew speakers who were asked to tell both false and true stories. Classification experiments on all possible combinations of these four feature sets using five supervised machine learning methods have been applied. The Part of Speech (POS) set was superior to all others and has been found as a key component. The best accuracy result (89.6%) has been achieved by a combination of sixteen POS-tags and one quantitative feature.

### 1 Introduction

"A lie is a false statement to a person or group made by another person or group who knows it is not the whole truth, intentionally" (Freitas-Magalhães, 2013). Dilmon (2014) defines a lie as "a linguistic message that conveys a falsehood or in which the truth is intentionally manipulated, in

order to arouse in the listener a belief which he would not otherwise have held."

The efforts to discover linguistic cues to detect lies are based on the assumption that there are differences between the language of an individual when he (or she) is not telling the truth and his (or her) "normal," truthful language. Fraser (1991) claims that these differences are the outcome of a feeling of stress, which is manifest in a decline in capacity for cognitive integration, in precision, in organization, and in ranking things. These difficulties result in a change in the normal elements of the speaker's language.

There were a few studies during the last four decades concerning verbal cues that characterize a lie discourse. Dulaney (1982) finds that the response time was shorter, there were fewer special words, a smaller number of verbs in the past tense, and a faster speech rhythm when an individual was lying; there were fewer words in the discourse, as well as a tendency to short messages. Knapp et al. (1974) find that there were more general declarations and fewer factual ones, linguistic ambiguity, repeated declarations, more markers of diminishment (few, a little, hardly) and fewer group markers (we, our, all of us), more markers of the other (they) and fewer personal declarations (I, me). Hollien and Rosenberg (1991) use lexical breakdown to investigate deception (type-token ratio - TTR), and finds less linguistic diversity when a person is practicing deception.

The studies of Dilmon (2007; 2008; 2012) conduct a comprehensive examination of the linguistic criteria that differentiate between the discourse of truth and of deception in the Hebrew language, and attempt to produce a primary test of the cognitive and emotional functions involved in the latter type of discourse. Forty three verbal criteria (Section 2.2) were classified according to the cognitive and emotional functions affecting the speaker, also addressing his level of awareness of these functions. Except one verbal criterion that was automatically computed by a program, the values of all other criteria for each story were computed by hand. This study starts from the end of the studies of Dilmon. Firstly, we implemented and/or applied four feature sets: POS-tag features, quantitative features, repetition features, and special expressions. Secondly, the application of the features is automatically done by a computer program in contrast to Dilmon's features (42 of her 43 features were computed by hand for each story). Thirdly, in contrast to Dilmon's studies that found which are the specific criteria that are statistically significant differentiators, we apply five supervised machine learning (ML) methods and various combinations of feature sets to find the best method for single-document classification, i.e., for each input story identifying whether it is a true or a false story. That will potentially lead to find discoveries concerning distinguishing between truth and false stories.

The task of distinguishing between true and false story as well as the interpretation of the obtained results are of practical interest for any language in general and for Hebrew in particular. Such a system can be of great help to the work of organizations, such as workplaces, detective agencies, police, and courts, to identify various types of stories.

The rest of this paper is organized as follows: Section 2 presents relevant background on linguistic examination in relevant systems, linguistic examination between discourses of truth and deception, text classification, and text classification of deception and true stories. Section 3 describes the classification model and the chosen feature sets. Section 4 presents the examined corpus, the experimental results and their analysis. Finally, Section 5 summarizes the main findings and suggests future directions.

## 2 Relevant Background

### 2.1 Linguistic examination in relevant systems

Argamon et al. (2009) describe an automatic process that profiles the author of an anonymous text. Accurate profiling of an unknown author is important for various tasks such as criminal investigations, market research, and national security. The deciphering the profile of someone is performed in the following way: Given a corpus of documents, marked as "male" and "female". Only four features were selected: sex, age, mother tongue, and neurotic level of disturbance behavior. Combination of linguistic features and various ML methods (Support vector machines and Bayesian regression) enable an automated system to effectively determine several such aspects of an anonymous author.

Chaski (2005) presents a computational, stylometric method that has obtained 95% accuracy and has been successfully used in investigating and adjudicating several crimes involving digital evidence. Chaski's approach focuses on language features that are easily achievable, e.g., word length, sentence length, word frequency, and the distribution of words according to different lengths.

Strous et al. (2009) describe an automatic process that characterizes and identifies schizophrenia in writing. This study investigates and analyzes computer texts written by 36 schizophrenia patients. Each document contains from 300 to 500 words. The system tested differences between these documents to documents written by people who are not sick with this disease. Observations have shown that methods using lexical and syntactic features obtained 83.3% accuracy. 60 features were chosen for the classification process: the 25 most frequent words in the corpus, the 20 most frequent letter tri-grams, and the average number of 15 repetitive words. The main conclusions are: (1) Some of the basic processes in schizophrenia are evident in writing; (2) Automatically identified characteristics of schizophrenic writing are closely related to the clinical description of the disorder; and (3) Automatic classification of samples in writing of schizophrenia is possible.

## 2.2 Linguistic examination between discourses of truth and deception

Hancock et al. (2005) found that “liars tended to produce more words, fewer first person singular but more third person pronouns, and more sense words than truth-tellers”. Only a small number of criteria were examined, the discourse being studied was written on a computer, the motivation to lie came from preliminary instructions, and the discourse examined was a conversation (not a full text).

The studies of Dilmon (2007; 2008; 2012) dealt with discovering linguistic differences between the discourse of truth and discourse of deception. Dilmon's studies present an investigation of 48 couples of stories told by 48 subjects. Each of them told a true story and a false story. The comparison was made using linguistic instruments, and the results obtained were examined statistically. The 48 subjects are native Hebrew speakers of both sexes, of different ages and a variety of backgrounds (with no criminal background).

The subjects were being instructed to take part in a game in which they had to tell two stories from their past, one true and the other an invention, and the “real” subjects would have to guess which of the stories was true and which an invention. In this way, the subjects themselves chose where and how they would mislead, and they would be motivated to provide stories that would make it hard to identify them as stories of deception. That is to say, they tried to escape detection, as would be the case in an actual deceptive situation. Apart from this instruction, they received no other instructions as to subject matter, length, or any other issue of the story's substance.

Dilmon (2012) compared between the true stories and the false stories. Her assumption is that the true stories indicate the subject's ordinary, “normal” language, while the false stories indicate deviations from that normal language. 43 criteria were defined by her to analyze the language of truth and falsity. Part of the criteria were translated to Hebrew from the foreign literature. Other criteria were collected after interviews with an attorney, a police investigator, a military police investigator, and two psychologists who had worked for the police. These criteria belong to the following areas: morphology, syntax, semantics, discourse analysis, and speech prosody.

42 out of 43 criteria were calculated manually. All these criteria were examined whether they differentiate between the discourse of truth and of deception. Statistical analyses using MANOVA were performed with repeated measures for each linguistic criterion. 19 criteria were found to differentiate significantly between the two types of discourse. 5 out of the 19 criteria that have been found as significant belong to the morphology area as follows: 1- # of past tense verbs, 2- # of present tense verbs, 3- # of future tense verbs, 4- # of first person verbs, and 5- # of third person verbs. All these criteria are normalized by the # of verbs in the tested story.

## 2.3 Text classification

Text classification (TC) is a supervised learning task that assigns natural language text documents to one or more predefined categories (Sebastiani, 2002). The TC task is one of the most fundamental tasks in data mining (DM) and machine learning (ML) literature (Aggarwal and Zhai, 2012).

TC has been applied in various domains, e.g., document indexing, document filtering, information retrieval (IR), information extraction (IE), spam filtering, text filtering, text mining, and word sense disambiguation (WSD) (Pazienza, 1997; Knight, 1999; Sebastiani, 2005).

There are two main types of TC: TC according to categories and to stylistic classification. TC according to categories (e.g., disciplines, domains, and topics) is usually based on content words and/or n-grams (Cavnar and Trenkle, 1994; Damashek, 1995; Martins and Silva, 2005; Liparas et al., 2014).

Literature documents, for instance, are different from scientific documents in their content words and n-grams. However, stylistic classification, e.g., authorship attribution (Stamatatos, 2009; Koppel et al., 2011), ethnicity/time/place (HaCohen-Kerner et al., 2010A; 2010B), genre (Stamatatos, 2000; Lim et al., 2005), gender (Hota et al., 2006; Koppel et al., 2002), opinion mining (Dave et al., 2003), computer science conference classification (HaCohen-Kerner et al., 2013), and sentiment analysis (Pang et al., 2002), is usually based on various linguistic features, such as function words, orthographic features, parts of speech (POS) (or syntactic) features, quantitative features, topographic features, and vocabulary richness.

## 2.4 TC of deception and true stories

Mihalcea and Strapparava (2009) present initial experiments in the recognition of deceptive language. They introduce three data sets of true and lying texts containing 100 true and 100 false statements for each dataset. They use two classifiers: Naïve Bayes and SVM. Their features were words belonging to several special word classes, e.g., friends (friend, companion, body), and self (our, myself, mine, ours). No feature selection was performed, and stopwords were not removed. Using a 10-fold cross-validation test their accuracy results were around 70%.

Ott et al. (2011) develop a dataset containing 400 truthful hotel reviews and 400 deceptive hotel reviews. Their features were Linguistic Inquiry and Word Count (LIWC) features extracted by the LIWC software (Pennebaker et al., 2007), relative POS frequencies extracted by the Stanford Parser (Klein and Manning, 2003) and 3 n-gram feature sets (unigrams, bigrams, and trigrams). Ott et al. show that the detection of deceptive opinion spam is well beyond the capabilities of human judges. Using Naïve Bayes and SVMlight (Joachims, 1999) and a 5-fold cross-validation test they have found that a bigram-based classification based on unigrams and bigrams obtained an accuracy of 89.6%, and a combination of LIWC features, unigrams and bigrams performed slightly better (89.8%).

## 3 The Classification Model and the Chosen Feature Sets

We decided to use Dilmon's stories as our data set. We defined, programmed and automatically calculated features for the input stories. In contrast to Dilmon, who calculated the ability of each feature alone to statistically distinguish between true and false stories, we investigated the ability of various combinations of features to classify between true and false stories using various ML methods.

### The main stages of the model are as follows:

1. Building a corpus containing 96 stories (48 false and 48 true stories).
2. Computing all four feature sets including the POS-tag features using the tagger built by Adler (Adler, 2007; Adler et al., 2008). This tagger achieved 93% accuracy for word segmentation and

POS tagging when tested on a corpus of 90K tokens.

3. Applying five ML methods for each possible combination of feature sets using default parameters.

4. Filtering out non-relevant features using InfoGain (IG) (Yang and Pedersen, 1997) and re-applying the best ML method found in stage #3.

### Features

In this paper, we consider 145 features divided into four meaningful linguistic feature sets as follows: 123 POS-tag features, 4 quantitative features, 9 repetition features, and 9 special expressions. These four feature sets have neither been defined nor applied by Dilmon. In this research, some of Dilmon's (2008) criteria have not been examined (e.g., discourse analysis and prosodic elements as stuttering and hesitation marks) because it was difficult to automatically detect them. However, features such as tense verbs and person verbs have been applied among the POS-tag feature set.

We did not choose the bag of words (BOW) or N-gram (which are usually the most frequent continuous sequences of N-grams) as features because they are too simple; they have less meaning and they can be partially seen as a black box. As an example of their low significance is the fact that the linear ordering of the N-grams within the text is ignored. That is to say, these representations are essentially independent of the sequence of words in the collection.

The first chosen feature set contains 123 POS-tag features automatically extracted by Adler's tagger for the Hebrew language (Adler, 2007; Adler et al., 2008). This set contains features, which belong to many feature sub-sets: 7 prefix types, 28 part-of-speech tags, 3 gender types, 5 number types, 4 person types, 3 status types, 7 tense types, 4 pronoun types, 8 named-entity types, 4 interrogative types, 3 prefix types, 15 punctuation types, 5 number suffix types, 4 person suffix types, 2 polarity types, 7 Hebrew verbal stem types, 3 conjunction types, 5 number types, 3 gender suffix types, and 3 quantifier types.

The second feature set is the quantitative set containing 4 types of average # of letters per word, average # of letters per sentence, average # of words per sentence, and TTR (the number of different word types in a text divided by the total number of word tokens).

The third feature set is the repetition features containing the following 9 features: normalized # of n-gram words (for n=1, 2, 3, 4) that repeat themselves in the same sentence, respectively, normalized # of ‘ha’ (i.e., “the”, the definite article in Hebrew), and normalized # of n-gram words (for n=1, 2, 3, 4) that repeat themselves in the entire text only once, twice, 3, or 4 times, respectively. The normalization is done by a division of the computed value to the number of word tokens in the document.

The fourth and the last feature set is the special expressions set that contains the normalized # of the following 9 features: intensifiers, minimizing markers, negative expressions, positive expressions, time expressions, expressions of doubt, Emotive words and words describing emotions, demonstrative pronouns, generalized words, ‘et’ (a term used to indicate a direct object), and ‘shel’ (i.e., of, belonging to).

#### 4 Corpus and Experimental Results

The examined corpus (supplied by Dilmon) contains 96 stories (48 false and 48 true stories) that were told by 48 native Hebrew speakers (23 men and 25 women) between the ages of 20 and 45. The reasons for relatively small number of subjects are: (1) The subjects did not receive payment for their participation; each one of them volunteered to participate. It is not easy to find many volunteers for such action. (2) The course of Dilmon's study included a recording of the stories, varying in length from five minutes to an hour. Then an accurate transcription of the stories was required (receiving over 100 pages of transcribed text) and a careful count of all the linguistic characteristics. Table 1 presents general information about this corpus.

Type of story	Total # of words	Avg. # of words per story	Median value of words per story	Std. of words per story
True	8722	181.7	155.5	145.03
False	6720	140	113.5	103.09

Table 1. General information about the corpus.

Five supervised ML methods including two decision tree methods have been selected. The accuracy rate of each ML method was estimated by a 10-fold cross-validation test. These ML methods

include SMO and Naïve Bayes (that were examined in the two previous studies about true/false classification mentioned in sub-section 2.4). The five applied ML methods are:

(1) Reduced Error Pruning (REP)-Tree is a fast decision tree learner, which builds a decision/regression tree using information gain/variance and prunes it using reduced-error pruning with back fitting (Witten and Frank, 2005). This algorithm sorts values for only numeric attributes. Missing values are dealt with by splitting the corresponding instances into pieces. Because the tree grows linearly with the size of the samples presented, and that, after a while, no accuracy is gained through the increased tree complexity, pruning becomes helpful if used carefully (Elomaa and Kääriäinen, 2001).

(2) J48 is an improved variant of the C4.5 decision tree induction (Quinlan, 1993; Quinlan, 2014) implemented in WEKA. J48 is a classifier that generates pruned or unpruned C4.5 decision trees. The algorithm uses greedy techniques and is a variant of ID3, which determines at each step the most predictive attribute, and splits a node based on this attribute. J48 attempts to account for noise and missing data. It also deals with numeric attributes by determining where thresholds for decision splits should be placed. The main parameters that can be set for this algorithm are the confidence threshold, the minimum number of instances per leaf and the number of folds for REP. As described earlier, trees are one of the easiest thing that could be understood because of their nature.

(3) Sequential Minimal Optimization (SMO; Platt 1998; Keerthi et al. 2001; Hastie and Tibshirani, 1998) is a variant of the Support Vectors Machines (SVM) ML method (Cortes and Vapnik 1995; Vapnik 2013). The SMO technique is an iterative algorithm created to solve the optimization problem often seen in SVM techniques. SMO divides this problem into a series of smallest possible sub-problems, which are then resolved analytically.

(4) Logistic regression (LR; Cessie et al., 1992) is a variant of a probabilistic statistical classification model that is used for predicting the outcome of a categorical dependent variable (i.e., a class label) based on one feature or more (Cessie et al., 1992; Landwehr et al., 2005; Sumner et al., 2005).

(5) Naïve Bayes (NB; John and Langley, 1995; McCallum and Nigam, 1998) is a set of probabilistic classifiers with strong (naive) independence assumptions between the features. The Naive Bayes Classifier method is usually based on the so-called Bayesian theorem (the current probability is computed based on a previous related probability) and is particularly suited when the number of the features is high.

These ML methods have been applied using the WEKA platform (Witten and Frank, 2005; Hall et al., 2009) using the default parameters. After finding the best ML method we have performed further experiments using only this method. Non-relevant features were filtered out using Information gain (InfoGain, IG), a feature selection metric for text classification. IG is a popular measure of feature goodness in text classification (Yang and Pedersen, 1997). It measures the number of bits of information obtained for category prediction by knowing the presence or absence of a feature. In their comparative study, Yang and Pedersen reported that IG and Chi performed best in their multi-class benchmarks. Forman (2003) reported that IG is the best filtering method when one is limited to 20-50 features. In Forman's experiments, IG dominates the performance of Chi for every size of the feature set. The accuracy of each ML method was estimated by a 10-fold cross-validation test.

Combinations of feature sets	Rep-Tree	J48	SMO	LR	NB
P	60.4	68.8	<b>80.2</b>	63.5	78.1
Q	61.5	61.5	67.7	64.6	66.7
R	52.1	57.3	60.4	61.5	61.5
S	68.8	66.7	77.1	77.1	68.8
P, Q	62.5	66.7	82.3	68.8	79.2
P, R	60.4	70.8	75.0	65.5	78.2
P, S	57.3	67.7	<b>83.3</b>	62.5	79.2
Q, R	60.4	67.7	63.5	67.7	67.7
Q, S	70.8	69.8	77.1	76.0	74.0
R, S	70.8	60.4	75.0	75.0	68.8
P, Q, R	62.5	67.7	77.1	67.7	79.2
P, R, S	71.9	65.6	81.3	71.9	79.2
P, Q, S	58.3	66.7	<b>84.4</b>	76.0	81.3
Q, R, S	68.8	66.7	75.0	76.0	70.8
P, Q, R, S	58.3	63.5	81.3	69.8	80.2

Table 2. Accuracy results for the classification of True/False stories.

In this research, there are four feature sets (section 3): POS-tags (P), Quantitative (Q),

Repetitions (R), and Special Expressions (S). Therefore, there are  $2^4 = 16$  combinations of feature sets (including the empty set). For each ML method we tried all 15 non-empty combinations of feature sets.

Table 2 presents the accuracy results for the classification of true/false stories according to all 15 combinations of feature sets. These results were obtained by applying the 5 supervised ML methods mentioned in Section 3.

Several general conclusions can be drawn from Table 2:

- The first 4 rows present the accuracy results using only one feature set. The best result for 3 ML methods (SMO, J48 and NB) was achieved by the POS-tags set. The best result out of these results was obtained by the POS-tags set using SMO. Similar to Ott et al. (2011) we related to the accuracy results achieved by the POS-tag features (80.2%) as the baseline with which to compare our other results.
- The POS-tags feature set (80.2%) is superior to the other single sets. Several possible explanations for this finding are: this set includes the largest number of features (123), and these features include widespread information about the whole text, which is relevant to the task at hand.
- The SMO method obtained the best accuracy result results for most of the set combinations (in 8 out of 15 experiments).
- The best accuracy result using a combination of 2 sets (83.3%) was obtained using a combination of the POS-tags and the special expressions.
- The best accuracy result in Table 2 (84.4%) was obtained using a combination of 3 sets: POS-tags, quantitative and the special expressions.
- The addition of the repetitions features to the 3 sets (i.e., the combination of all 4 sets) led to a decline in the results (81.3%). The repetitions set was the set with the worst results compared with the other sets for all five ML methods.
- The improvement rate from the best set to the best combination of sets is 4.2%.

Since SMO has been found as the best ML method for our classification task, we decided to do further experiments using only SMO and IG (as explained above).

Combinations of feature sets	SMO before IG		SMO after IG							
	# of feat.	Acc.	# of feat.	Acc.	True			False		
					P	R	F	P	R	F
P	123	80.2	15	86.5	80.9	79.2	80.0	79.6	81.3	80.4
Q	4	67.7	1	57.3	68.4	27.1	38.8	54.5	87.5	67.2
R	9	60.4	1	53.1	53.1	54.2	53.6	53.2	52.1	52.6
S	9	77.1	4	69.8	67.3	77.1	71.8	73.2	62.5	67.4
P, Q	127	82.3	17	89.6	91.3	87.5	89.4	88.0	91.7	89.8
P, R	132	75.0	17	87.5	90.9	83.3	87.0	84.6	91.7	88.0
P, S	132	83.3	20	85.4	88.6	81.3	84.8	82.7	89.6	86.0
Q, R	13	63.5	2	63.5	65.1	58.3	61.5	62.3	68.8	65.3
Q, S	14	77.1	5	78.1	77.6	79.2	78.4	78.7	77.1	77.9
R, S	13	75.0	4	69.8	67.3	77.1	71.8	73.2	62.5	67.4
P, Q, R	136	77.1	18	88.5	91.1	85.4	88.2	86.3	91.7	88.9
P, R, S	136	81.3	20	85.4	88.6	81.3	84.8	82.7	89.6	86.0
P, Q, S	137	84.4	21	89.6	93.2	85.4	89.1	86.5	93.8	90.0
Q, R, S	22	75.0	6	79.2	78.0	81.3	79.6	80.4	77.1	78.7
P, Q, R, S	145	81.3	22	89.6	93.2	85.4	89.1	86.5	93.8	90.0

Table 3. Accuracy results for combinations of feature sets using SMO and IG.

Table 3 presents the accuracy results for all combinations of feature sets using SMO (the best ML method according to Table 2) before and after filtering out non-relevant features using IG. In addition, for the stage after activating IG we also present the precision, recall, and F-score results for each type of story (true, false) for all possible combinations of the four feature sets. The following conclusions can be drawn from Table 3 regarding the classification of True/False stories using SMO and IG:

- The best accuracy result (89.6%) has been achieved by three different combination sets. The combination with the smallest number of feature sets, is the combination of two sets: POS-tag and quantitative, which contains 17 features including 16 POS-tag features and one quantitative feature.

- The improvement rate of this combination of two sets from the initial state before performing IG to the state after performing IG is 7.3%. This improvement has been achieved due to the filtering out of 110 features out 127!

- The relatively similar accuracy, precision, recall, and F-score results for both types of stories (true, false) for all types of set combinations represent that the classification results are at the same level of quality for both types of stories.

- By looking at the results of the best combinations in Table 3 (colored with red and blue), we see that on the one hand, the precision values are higher for the true stories (i.e., less false

positives; which means that the system has a high ability to present only relevant true stories), and on the other hand, the recall values are higher for the false stories (i.e., less false negatives; which means that the system has a high ability to present all relevant false stories)

Detailed results for the best combination (16 POS-tag features and one quantitative feature) using SMO and IG are presented in Tables 4 and 5. Table 4 presents the suitable confusion matrix and Table 5 shows the values of the ROC and PRC areas. The area under the ROC curve (Bradley 1997; Fawcett 2006) and the area under the PRC curve, i.e., the area under the precision-recall curve (Boyd et al., 2013) are often used to evaluate the performance of ML methods.

		Actual answer	
		True	False
Classifier's answer	True	TP=44	FP=4
	False	FN=6	TN=42

Table 4. The confusion matrix.

	True	False
ROC area	89.6	89.6
PRC area	86.1	84.8

Table 5. The ROC and PRC areas.

Using the TP, FP, FN, FP, and TN values in the confusion matrix, are computed the four popular measures: recall, precision, accuracy and f-measure (Table 3). The ROC area is around 90% and the PRC area is around 85%-86% indicating very good classification performance of the SMO method using the 17 chosen features.

Another deeper observation shows several interesting findings about the most distinguishing features according to the IG method (i.e., features that received the highest weights). Table 6 presents some distinguishing POS features.

Distinguishing POS features	Finding	Meaning
Person-1 (first person)	The average of this feature for the true stories is significantly higher	Truthful people use relatively more first person pronouns
Person-3 (third person)	The average of this feature for the false stories is significantly higher	Liars use relatively more third person pronouns
POS-negation (negation words)	The average of this feature for the false stories is significantly higher	Liars use relatively more negation words

Table 6. Distinguishing POS features according to SMO and IG.

Our findings concerning the use of first-person pronouns, and negative words are consistent with the conclusions of Hancock et al. (2005) who found that the discourse of deception used fewer first-person pronouns, and more negative words. Our findings concerning use of first and third person pronouns are also consistent with the conclusions of Knapp et al. (1974) who found that a lie discourse contains more markers of the other and fewer personal declarations (I, me).

Furthermore, our findings are also consistent with some of Dilmon (2008): (1) The use of negative words in the false stories might reveal the speaker's negative attitude toward his invention, and his insecurity from being in the position of misleading the listener, and (2) Higher use of verbs in the third person and minimal use of verbs in the first person in false stories may imply the speaker's desire to distance himself from a description of the event and from the possibility of accepting responsibility for his actions.

From a pragmatic standpoint, a deception is a deviation from Grice's (1975) "Cooperative Principle", which is subdivided into 4 maxims: of quantity, of quality, of relation, and of manner. He stresses that the meticulous observance of the maxim of quality is a fundamental pre-condition that ensure the operation of the other maxims. Mey (2001) claims that concealment techniques (e.g., deliberate omission, and uninformative or disinformative remarks) contradict the Cooperative Principle of Grice. By using negative words and third person verbs, the speaker is violating the maxim of quality.

## 5 Summary and Future Work

In this paper, we present a methodology for distinguishing between true and false stories based on various linguistic features. The POS-tag set containing 123 features was superior to all other sets with an accuracy result of 80.2%. The best accuracy result (89.6%) was obtained by SMO and IG using two feature sets including only sixteen POS-tag features and one quantitative feature. These results suggest that stylistic differences between any types of true and false stories can be quantified along the lines presented in this paper.

The main contribution of this research is the careful feature set engineering based on analyses construction of feature sets derived from previous studies. This together with the competition between five well-known supervised ML methods, and filtering out of non-relevant features using IG for SMO (the best found ML method), yields considerably improved accuracy results.

Future research proposals are: (1) Apply this classification model to other types of true and false stories coming from other domains and written in various languages; (2) Implement feature sets with a focus on special compound linguistic features that differentiate between true and false stories, speech features such as hesitations or repetitions, n-gram features and other types of stylistic feature sets; (3) Perform experiments to see if some interactions at feature level, not feature set level, have any impact on the classification accuracy; and (4) Perform experiments of distinguishing between true and false stories by people, and comparing their results versus those performed by our system.

## References

- Meni Adler. 2007. Hebrew Morphological Disambiguation: An Unsupervised Stochastic Word-based Approach, Ph.D. Dissertation, Ben Gurion University, Israel.
- Meni Adler, Yael Netzer, David Gabay, Yoav Goldberg, and Michael Elhadad. 2008. Tagging a Hebrew Corpus: The Case of Participles, In *Proceedings of the LREC-2008*, European Language Resources Association, Marrakech, Morocco.
- Charu C. Aggarwal and ChengXiang Zhai. 2012. Mining text data. New York, NY: Springer.
- Kendrick Boyd, Kevin H. Eng, and C. David Page. 2013. Area under the Precision-Recall Curve: Point Estimates and Confidence Intervals. In *Proceedings of the Machine learning and knowledge discovery in databases*, pages 451-466. Springer Berlin Heidelberg.
- Andrew P. Bradley. 1997. The use of the Area under the ROC Curve in the Evaluation of Machine Learning Algorithms. *Pattern Recognition* 30: 1145–1159. doi: 10.1016/S0031-3203(96)00142-2
- Leo Breiman. 2001. Random Forests. *Machine Learning*, 45(1): 5–32.
- William B. Cavnar and John M. Trenkle. 1994. N-Gram-Based Text Categorization. *Ann Arbor MI*, 48113(2): 161-175.
- Renato F. Corrêa and Teresa B. Ludermir. 2002. Automatic Text Categorization: Case Study, In *Proceedings of the VII Brazilian Symposium on Neural Networks*, SBRN 2002, page 150, IEEE
- Corinna Cortes and Vladimir Vapnik. 1995. Support-vector Networks. *Machine learning*, 20(3): 273–297.
- Marc Damashek. 1995. Gauging Similarity with N-grams: Language-independent Categorization of Text, *Science*, 267(5199): 843-848.
- Kushal Dave, Steve Lawrence, and David M. Pennock. 2003. Mining the Peanut Gallery: Opinion Extraction and Semantic Classification of Product Reviews. In *Proceedings of the 12th international conference on World Wide Web*, pages 519-528, ACM.
- Rakefet Dilmon. 2004. Linguistic Differences between Lie and Truth in Spoken Hebrew – Doctoral Dissertation. Bar Ilan University, Ramat Gan, Israel (in Hebrew).
- Rakefet Dilmon. 2007. Fiction or Fact? Comparing True and Untrue Anecdotes, *Hebrew Linguistics*, 59: 23-42 (in Hebrew).
- Rakefet Dilmon. 2008. Between Thinking and Speaking - Linguistic Tools for Detecting a Fabrication, *Journal of Pragmatics*, 41(6): 1152-1170.
- Rakefet Dilmon. 2012. Linguistic Examination of Police Testimony – Falsehood or Truth? In: R. Peled-Laskov, E. Shoham & M. Carmon (eds.), False Convictions: Philosophical, Organizational and Psychological Aspects (433 pages), Perlstein-Ginosar & Ashkelon Academic College, Tel-Aviv, pages 95-114 (in Hebrew).
- Rakefet Dilmon. 2013. False speech, linguistic aspects in Hebrew, in: D. A. Russell, G. Khan, & D. L. Vanderzwaag (eds.), *The Encyclopedia of Hebrew Language and Linguistics*, Leiden, Brill, Boston and Tokyo, pages 542-546.
- Earl F. Dulaney. 1982. Changes in Language Behavior as a Function of Veracity. *Human Communication Research*, 9(1): 75-82.
- Tapio Elomaa, Matti Kääriäinen. 2001. An Analysis of Reduced Error Pruning. *Journal of Artificial Intelligence Research* 15: 163–187. doi: 10.1613/jair.816
- Tom Fawcett, 2006. An Introduction to ROC Analysis. *Pattern Recognition Letters*, 27(8): 861–874. doi: 10.1016/j.patrec.2005.10.010
- George Forman. 2003. An Extensive Empirical Study of Feature Selection Metrics for Text Classification, *The Journal of machine learning research*, 3: 1289-1305.
- Bruce Fraser. 1991. Questions of Witness Credibility. Working Papers Series, Program on Negotiation. Cambridge, MA: Harvard Law School, pages 3-91.
- Armindo Freitas-Magalhães. 2013. The Face of Lies. Porto: FEELab Science Books. ISBN 978-989-98524-0-2.
- Herbert P. Grice. 1975. Logic and conversation. In Cole Peter & Jerry L. Morgan (Eds.), *Syntax and semantics*, vol. 3 Speech acts: 41-58. New York: Academic Press.
- Yaakov HaCohen-Kerner, Hananya Beck, Elchai Yehudai, and Dror Mughaz. 2010A. Stylistic Feature Sets as Classifiers of Documents According to their Historical Period and Ethnic Origin. *Applied Artificial Intelligence*, 24(9): 847-862.
- Yaakov HaCohen-Kerner, Hananya Beck, Elchai Yehudai, Mordechai Rosenstein, and Dror Mughaz. 2010B. Cuisine: Classification using Stylistic Feature Sets and/or Name-Based Feature Sets. *Journal of the*

- American Society for information Science & Technology (JASIST)*, 61(8): 1644-1657.
- Yaakov HaCohen-Kerner, Avi Rosenfeld, Maor Tzidkani, and Daniel Nisim Cohen. 2013. Classifying Papers from Different Computer Science Conferences. In *Proceedings of the Advanced Data Mining & Applications. ADMA 2013, Part I, LNAI 8346*, pages 529-541, Springer Berlin Heidelberg.
- Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, and Ian H. Witten. 2009. The WEKA Data Mining Software. *ACM SIGKDD Explorations Newsletter*, 11(1):10.
- Jeffrey T. Hancock, Lauren Curry, and Saurabh Goorha, Michael Woodworth. 2005. Automated Linguistic Analysis of Deceptive and Truthful Synchronous Computer-Mediated Communication. In *Proceedings of the 38th Hawaii International Conference on System Science*, pages 1-10.
- Trevor Hastie and Robert Tibshirani. 1998. Classification by Pairwise Coupling. *The annals of statistics*, 26 (2): 451-471.
- Harry Hollien and Aaron E. Rosenberg. 1991. *The Acoustics of Crime: The new Science of Forensic Phonetics*. New York: Plenum.
- Sobhan R. Hota, Shlomo Argamon, and Rebecca Chung. 2006. Gender in Shakespeare: Automatic Stylistics Gender Character Classification Using Syntactic, Lexical and Lemma Features. *Digital Humanities and Computer Science (DHCS)*.
- George H. John, Pat Langley. 1995. Estimating Continuous Distributions in Bayesian Classifiers. In *Proceedings of the 11th Conference on Uncertainty in Artificial Intelligence*, San Mateo, pages 338-345.
- Thorste Joachims. 1999. Making Large Scale SVM Learning Practical. In *Advances in kernel methods*, page 184. MIT Press.
- S. Sathiya Keerthi, Shirish K. Shevade, Chiranjib Bhattacharyya, and Karaturi R. K. Murthy. 2001. Improvements to Platt's SMO Algorithm for SVM Classifier Design. *Neural Computation*, 13 (3): 637-649.
- Dan Klein and Christopher D. Manning. 2003. Accurate Unlexicalized Parsing. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics*, Volume 1: 423-430. Association for Computational Linguistics.
- Kevin Knight. 1999. Mining Online Text, *Commun. ACM*, 42(11): 58-61.
- Moshe Koppel, Shlomo Argamon, and Anat R. Shimoni. 2002. Automatically Categorizing Written Texts by Author Gender. *Lit Linguist Computing*, 17 (4): 401-412.
- Moshe Koppel, Jonathan Schler, and Shlomo Argamon. 2011. Authorship Attribution in the Wild. *Language Resources & Evaluation*, 45(1): 83-94.
- Mark L. Knapp, Roderick P. Hart, and Harry S. Dennis. 1974. An Exploration of Deception as a Communication Construct. *Communication Research*, 1: 15-29. doi:10.1111/j.1468-2958.1974.tb00250.x
- Niels Landwehr, Mark Hall, and Eibe Frank. 2005. *Logistic Model Trees*. *Machine Learning*, 59 (1-2): 161-205.
- Chul S. Lim, Kong J. Lee, and Gil C. Kim. 2005. Multiple Sets of Features for Automatic Genre Classification of Web Documents. *Information processing & management* (5): 1263-1276.
- Dimitris Liparas, Yaakov HaCohen-Kerner, Stefanos Vrochidis, Anastasia Moutzidou, and Ioannis Kompatsiaris. 2014. News Articles Classification Using Random Forests and Weighted Multimodal Features. In *Multidisciplinary Information Retrieval, Proceedings of the 7th Information Retrieval Facility Conference*, pages 63-75. Springer International Publishing.
- Martins, B., Silva M. J. 2005. Language Identification in Web Pages. In *Proceedings of the 2005 ACM symposium on applied computing*, pages 764-768. ACM
- Jacob Mey. 2001. *Pragmatics: An introduction*. Malden, MA: Blackwell Publishers.
- Andrew McCallum and Kamal Nigam. 1998. A Comparison of Event Models for Naive Bayes Text Classification. In *Proceedings of the AAAI-98 workshop on learning for text categorization*, Vol. 752, pages 41-48.
- Rada Mihalcea, and Carlo Strapparava. 2009. The Lie Detector: Explorations in the Automatic Recognition of Deceptive Language. In *Proceedings of the ACL-IJCNLP 2009 Conference*, Short Papers, pages 309-312. Association for Computational Linguistics.
- Myle Ott, Yejin Choi, Claire Cardie, Jeffrey T. Hancock. 2011. Finding Deceptive Opinion Spam by any Stretch of the Imagination. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, Volume 1, pages 309-319. Association for Computational Linguistics.
- Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan. 2002. Thumbs up?: Sentiment Classification using

- Machine Learning Techniques. In *Proceedings of the ACL-02 conference on Empirical methods in natural language processing (EMNLP'02)*, Volume 10, pages 79-86.
- Maria T. Pazienza. (ed.) 1997. Information Extraction: A Multidisciplinary Approach to an Emerging Information Technology. Springer International Publishing.
- James W. Pennebaker, Cindy K. Chung, Molly Ireland, Amy Gonzales, and Roger J. Booth. 2007. The Development and Psychometric Properties of LIWC2007.
- John C. Platt. 1998. Sequential Minimal Optimization: A Fast Algorithm for Training Support Vector Machines. *Advances in Kernel Methods Support Vector Learning*, 208:1-21.
- J. Ross Quinlan. 1993. Programs for Machine Learning. Volume 240. Elsevier.
- J. Ross Quinlan. 2014. C4. 5: Programs for Machine Learning. Elsevier.
- Fabrizio Sebastiani. 2002. Machine Learning in Automated Text Categorization. *ACM computing surveys (CSUR)*, 34 (1): 1-47.
- Fabrizio Sebastiani. 2005. Text Categorization, pages 683-687. Retrieved from: <http://nmis.isti.cnr.it/sebastiani/Publications/EDTA05.pdf>
- Efstathios Stamatatos, Nikos Fakotakis, and George Kokkinakis. 2000. Automatic Text Categorization in Terms of Genre and Author. *Comput. Linguist*, 26 (4): 471-495.
- Efstathios Stamatatos. 2009. A Survey of Modern Authorship Attribution Methods. *Journal of the American Society for information Science & Technology (JASIST)*, 60 (3): 538-556.
- Marc Sumner, Eibe Frank, and Mark Hall. 2005. Speeding up Logistic Model Tree Induction. In *Proceedings of the Knowledge Discovery in Databases, PKDD 2005*, pages 675-683, Springer Berlin Heidelberg.
- Vladimir Vapnik. 2013. The Nature of Statistical Learning Theory. Springer Science & Business Media.
- Ian H. Witten and Eibe Frank, E. 2005. Data Mining: Practical Machine Learning Tools & Techniques, Second Edition (Morgan Kaufmann Series in Data Management Systems). San Mateo, CA: Morgan Kaufmann.
- Yiming Yang. 1999. An Evaluation of Statistical Approaches to Text Categorization. *Information Retrieval*, 1(1-2): 69-90.
- Yiming Yang and Jan O. Pedersen. 1997. A Comparative Study on Feature Selection in Text Categorization. In *Proceedings of the 14th International Conference on Machine Learning*, 97: 412-420.

## Bilingually motivated segmentation and generation of word translations using relatively small translation data sets

K. M. Kavitha<sup>1,3</sup>   Luís Gomes<sup>1,2</sup>   José Gabriel P. Lopes<sup>1,2</sup>

<sup>1</sup>NOVA Laboratory for Computer Science and Informatics (NOVA LINCS)

Faculdade de Ciências e Tecnologia, Universidade Nova de Lisboa

2829-516 Caparica, Portugal.

luismsgomes@gmail.com   gpl@fct.unl.pt

<sup>2</sup>ISTRION BOX-Translation & Revision, Lda., Parkurbis, Covilhã 6200-865 Portugal.

<sup>3</sup> Department of Computer Applications, St. Joseph Engineering College

Vamanjoor, Mangaluru, 575 028, India.

kavitham@sjec.ac.in

### Abstract

Out-of-vocabulary (OOV) bilingual lexicon entries is still a problem for many applications, including translation. We propose a method for machine learning of bilingual stem and suffix translations that are then used in deciding segmentations for new translations. Various state-of-the-art measures used to segment words into their sub-constituents are adopted in this work as features to be used by an SVM based linear classifier for deciding appropriate segmentations of bilingual pairs, specifically, in learning bilingual suffixation.

### 1 Introduction

OOV bilingual lexicon entries still remain an open problem and the approach proposed in this paper will contribute to solve this by machine learning of bilingual stem and suffix pairs using a very small English (EN)-Hindi (HI) bilingual lexicon. These bilingual segments are then used in deciding segmentations for unseen translations and also in generating new ones. We examine a combination of commonly used segmentation measures as clues for bilingual suffixation of unseen translations in a minimally supervised framework.

No translation extraction technique guarantees the extraction of all possible translation pairs specially when not found or are infrequent in parallel corpora. Source-target asymmetry further adds to the problem for morphologically poor and rich language pairs, as is the case of English and Hindi. Noun, verb or adjective forms in English tend to have multiple translations in Hindi. Consider the English term ‘good’ with 3 possible translations: ‘acChA’, ‘acChI’ and ‘acChe’ in Hindi. Each of these represent

variants of the basic word form ‘acChA’, where ‘-A’ and ‘-I’ represent singular masculine and feminine, while ‘-e’ denotes a plural adjective suffix. As all the forms might hardly be seen in the training data, there is a need for identifying morphological similarities in the known example pairs<sup>1</sup>. In the referred example, the three word forms share the stem ‘acCh’ and differ in the endings ‘-A’, ‘-I’, ‘-e’. As these inflections appear as endings for other words they serve in identifying word classes. Thus, the separation of morphological suffixes conflates various forms of a word, into a stem which is a crucial source of information. On the other hand, suffixes that occur frequently with words belonging to similar class, could be utilised for generating unknown forms. Hence, by using the morphological information, all possible forms can be inferred by combining different component morphemes from different mappings learnt using the example pairs in the translation lexicon.

We discuss a generative approach for suggesting new translations based on the morphological similarities learnt from translation examples seen in the existing bilingual lexicon. The approach is distinguishing as we rely on the frequent forms (suffixes) occurring in translations rather than on words in just one language. Fundamental to this generation strategy, we have 2 phases involving learning and classification. Firstly, the bilingual approach to learning morph-like units is used in preparing the training data (Mahesh et al., 2014). This involves identification and extraction of orthographically and semantically similar bilingual segments (as for instance, ‘good’ ⇔ ‘acCh’) occurring in known translation

<sup>1</sup>Words consist of high-frequency strings (affixes) attached to low-frequency strings (stems) (Hammarström, 2009)

examples (*'acChA'*, *'acChI'* and *'acChe'*), together with their bilingual extensions constituting dissimilar bilingual segments (bilingual suffixes) ( $' \Leftrightarrow 'A' \mid 'e' \mid 'I'$ )<sup>2</sup>. The common part of translations that conflates all its bilingual variants<sup>3</sup> represents a bilingual stem (*'good'*  $\Leftrightarrow$  *'acCh'*). The different parts of the translations contributing to various surface forms represent bilingual suffixes or bilingual morphological extensions ( $' \Leftrightarrow 'A' \mid 'e' \mid 'I'$ ). Further, bilingual suffixes representing bilingual extensions for a set of bilingual stems form bilingual suffix classes<sup>4</sup>, hence allowing safer translation generalisation. The bilingual suffix classes thus learnt along with the bilingual lexicon constitutes the training data set for the classification phase. Upon identification of the segmentation boundary (by classification), depending on the bilingual suffix and the stem surfaced for the given unseen translation, the bilingual pair is then classified into one of the bilingual suffix classes identified in the training phase. New translations are then suggested by simple concatenation of bilingual stems and suffixes belonging to the identified class.

## 2 Related Work

### 2.1 Monolingual Approaches

Lexical inference or morphological processing techniques have been established for handling unknown terms that are variations of known forms. Moreover, learning suffixes and suffixation operations from the corpus or lexicon of a language allows new words to be generated. Such approaches are categorised as supervised (Déjean, 1998), semi-supervised (Lindén et al., 2009) and unsupervised (Goldsmith, 2001; Creutz and Lagus, 2005; Monson et al., 2009).

The state-of-the-art approaches to unsupervised morphology learning are overviewed and discussed with sufficient level of detail by Wicentowski and Yarowsky (2002) and Hammarström and Borin (2011), respectively. A most recent work integrates orthographic and semantic view of words and models word formation in terms of morphological chains (Narasimhan et al., 2015). To address the morphological segmentation problem, Kirschenbaum (2015) suggests the use of segmentations de-

rived from words sharing similar distribution and form in analysing less frequent words.

Partially supervised strategies for morphology learning may be viewed as classification tasks. The classifier trained on known paradigms classifies the unseen words into paradigms or induces new paradigms (Lindén et al., 2009).

### 2.2 Bilingual Learning Approaches

Sasaoka *et al.*, proposed bilingual inductive learning mechanism for predicting translations for unknown words (Sasaoka et al., 1997). Common and different parts of strings between known words and their translations represent the example strings, referred as Piece of Word (PW) and Pair of Piece of Word (PPW). The bilingual pairs of these extracted example strings maintained as a Pair of Piece of Word (PPW) dictionary form the basis of the prediction process.

Snyder and Barzilay (2008) proposed simultaneous morphology learning for discovery of abstract morphemes using multiple languages. To boost the segmentation decisions, Poon et al. (2009), proposed discriminative log-linear model employing overlapping contextual features.

In our previous work, we proposed an approach for learning bilingual suffixation operations by utilising the translation lexicon as a parallel resource (Mahesh et al., 2014). As a pre-phase to translation generation, bilingual morph-like units conflating various translation forms are learnt and consequently clustered into bilingual suffix classes. Frequent forms occurring in translations rather than in word forms (in a language) are used in arriving at the segmentation decision. The ambiguities and complexities in decompositions are reduced as the translation forms impose a restricted subset over the entire universe of word forms from which segmentation decisions are made. Similar to the approach proposed by Sasaoka *et al.*, (Sasaoka et al., 1997), our approach (Mahesh et al., 2014) that we adapt here for preparing the partial training data, allows identification of common (bilingual stems) and different (bilingual suffixes) bilingual segments occurring in translation examples, which are then used in generating new translations.

## 3 Proposed approach

Much of the research ranging from text analysis for acquisition of morphology, to learning suffixes and

<sup>2</sup>Note the null suffix in the English side corresponding to gender and number suffixes in the Hindi side.

<sup>3</sup>Translations that are lexically similar.

<sup>4</sup>A *suffix class* may or may not correspond to Part-of-Speech such as noun or adjective but there are cases where the same suffix class aggregates nouns, adjectives and adverbs.

suffixation operations for partially overcoming OOV bilingual entries and generating necessary trustable bilingual entries, is driven by the fact that word is made up of high-frequency affixes attached to low-frequency stems (Hammarström, 2009). Extending this observation, we interpret a bilingual pair to be constituted by frequent bilingual suffixes attached to less frequent bilingual stems. The proposed approach operates in 2 stages: the *learning phase* for identifying bilingual suffix classes that partially serves as the training data and the *classification phase* for deciding segmentation.

### 3.1 Learning Phase

Learning bilingual segments using translation variants and their mapping into morphologically related classes closely follows the bilingual learning approach and involves learning bilingual suffixes and suffixation operations (Mahesh et al., 2014) (refer Algorithm 1).

**Definitions** Let  $L$  be a Bilingual Lexicon.

Let  $L_1, L_2$  be languages with alphabet set  $\Sigma_1, \Sigma_2$ .

$T = \{(w_{L_1}, w_{L_2}) | (w_{L_1}, w_{L_2}) \subset L\}$  be set of valid bilingual pairs (translations) in  $L$ .

$S = \{p_{i_{L_1}}, s_{i_{L_1}}, p_{i_{L_2}}, s_{i_{L_2}} | p_{i_{L_1}} s_{i_{L_1}} = w_{i_{L_1}};$

$p_{i_{L_2}} s_{i_{L_2}} = w_{i_{L_2}}; p_{i_{L_1}}, s_{i_{L_1}} \in \Sigma_1, p_{i_{L_2}}, s_{i_{L_2}} \in \Sigma_2\}$  be the set of substrings of  $w_{i_{L_1}}, w_{i_{L_2}}$ , where  $p_{i_{L_1}} s_{i_{L_1}}$  denotes the concatenation of stem  $p_{i_{L_1}}$  and suffix  $s_{i_{L_1}}$  in languages  $L_1$  and  $L_2$ .

Let  $S_{SuffixPair}$  be the set of bilingual suffix pairs and  $S_{StemPair}$  be the set of bilingual stem pairs.

Two translations  $(w_{1_{L_1}}, w_{1_{L_2}})$  and  $(w_{2_{L_1}}, w_{2_{L_2}}) \in L$  are said to be *similar* if  $|lcp(w_{1_{L_1}}, w_{2_{L_1}})| \geq 3$  and  $|lcp(w_{1_{L_2}}, w_{2_{L_2}})| \geq 3$ , where  $lcp$  is the longest common prefix of the strings under consideration.

#### Input - Bilingual/Translation Lexicon (L):

Translation lexicon refers to a dictionary which contains a term (taken as a single word - any contiguous sequence of characters) in the first language cross-listed with the corresponding term in the second language such that they share the same meaning or are usable in equivalent contexts. In Table 1, sample entries illustrate bilingual variants: *noun\_singular* forms (columns 1, 2 in 1<sup>st</sup> 7 rows) – *noun\_plural* forms (column 3, 4 in 1<sup>st</sup> 7 rows) and *adjective* forms (columns 1, 2 in last 4 rows) – *adverb* forms (columns 3, 4 in last 4 rows).

#### Output :

**List of Bilingual stem and suffix pairs:** These

Term (EN)	Term (HI)	Term (EN)	Term (HI)
process	प्रक्रिया (prakriyA)	processes	प्रक्रियाओं (prakriyAoM)
proof	प्रमाण (pramAN)	proofs	प्रमाणों (pramANoM)
plant	पौधा (pauDhA)	plants	पौधों (pauDhoM)
proceeding	कार्यवाही (kAryavAhi)	proceedings	कार्यवाहियों (kAryavAhiyoM)
plan	योजना (yojanA)	plans	योजनाएँ (yojanAeM)
prayer	प्रार्थना (prArthanA)	prayers	प्रार्थनाएँ (prArthanAeN)
promise	वाद (vAd)	promises	वादे (vAde)
usual	सामान्य /साधारण (sAmAny/sAdharaN)	usually	सामान्यतः/साधारणतः (sAmAnyatH/sAdharaNatH)
chief	प्रधान (pradhAn)	chiefly	प्रधानतः (pradhanataH)
rapid	शीघ्र (shighr)	rapidly	शीघ्रता (shighrata)
weak	दुर्बल (durbal)	weakly	दुर्बलता (durbalata)

Table 1: Bilingual variants in EN-HI Lexicon

include the list of bilingual stems (columns 3, 4 in Table 5) and suffixes (Table 4) with their observed frequencies in the training dataset. Sample bilingual stems include ‘*plant*’  $\Leftrightarrow$  ‘*pauDh*’, ‘*boy*’  $\Leftrightarrow$  ‘*laDak*’. Sample bilingual suffixes are (‘, ‘*I*’), (‘, ‘*A*’), (‘*ion*’, ‘*A*’) and are attached to 10,743, 29,529 and 457 different bilingual pairs respectively. These lists aid in identifying bilingual stems and bilingual suffixes, given a new translation.

#### Bilingual suffixes grouped by bilingual stems:

This represents which set of bilingual suffixes attach to which bilingual stem. In Table 2<sup>5</sup>, the bilingual suffixes, (‘*s*’, ‘*oM*’) and (‘*ous*’, ‘*I*’) attach to the same bilingual stem (‘*mountain*’, ‘*pahAD*’) yielding the surface forms ‘*mountains*’  $\Leftrightarrow$  ‘*pahADoM*’ and ‘*mountainous*’  $\Leftrightarrow$  ‘*pahADI*’.

Bilingual Stems	Bilingual Suffixes
(‘nation’, ‘राष्ट्र’) : (‘al’, ‘ीय’), (‘alism’, ‘ीयता’), (‘ality’, ‘ीयता’), (‘alist’, ‘ीयतावादी’)	
(‘nation’, ‘rAshTr’) : (‘al’, ‘Iya’), (‘alism’, ‘IyatA’), (‘ality’, ‘IyatA’), (‘alist’, ‘IyatAvAdI’)	
(‘mountain’, ‘पहाड़’) : (‘s’, ‘ोँ’), (‘ous’, ‘ी’)	
(‘mountain’, ‘pahAD’) : (‘s’, ‘oM’), (‘ous’, ‘I’)	

Table 2: Bilingual suffixes grouped by bilingual stems

**Bilingual Suffix Classes:** A set of bilingual stems that share same suffix transformations form a cluster or a bilingual suffix class. In the 1<sup>st</sup> row of Table 5, (‘, ‘*A*’) and (‘*s*’, ‘*oM*’) represent bilingual suffixes that combine with bilingual stems, ‘*plant*’  $\Leftrightarrow$  ‘*pauDh*’, ‘*boy*’  $\Leftrightarrow$  ‘*laDak*’ and many more. These allow new translation forms to be subsequently suggested upon identification of bilingual stems and suffixes in an unseen translation given as input.

<sup>5</sup>2<sup>nd</sup> line in each row shows transliterations for HI terms

**Algorithm 1** Learning Bilingual Suffix Classes

---

```

1: procedure LEARNBILINGUALSUFFIXCLASS
2:   for each translation  $(a_{L1}, a_{L2}) \in L$  do
3:     if  $\exists (b_{L1}, b_{L2})$  similar to  $(a_{L1}, a_{L2})$ , and  $(c_{L1}, c_{L2})$  similar to  $(d_{L1}, d_{L2}) \in L$ ,
4:       where  $p_{1L1}, p_{1L2}, p_{2L1}, p_{2L2}, s_{1L1}, s_{1L2}, s_{2L1}, s_{2L2} \in S$ , and
5:        $(a_{L1}, a_{L2}) = ((p_{1L1} s_{1L1}), (p_{1L2} s_{1L2}))$ ;  $(b_{L1}, b_{L2}) = ((p_{1L1} s_{2L1}), (p_{1L2} s_{2L2}))$ ,
6:        $(c_{L1}, c_{L2}) = ((p_{2L1} s_{1L1}), (p_{2L2} s_{1L2}))$ ;  $(d_{L1}, d_{L2}) = ((p_{2L1} s_{2L1}), (p_{2L2} s_{2L2}))$  then
7:         add  $(p_{1L1}, p_{1L2})$  to the list of bilingual stems  $S_{StemPair}$ .
8:         add  $((s_{1L1}, s_{1L2}), (s_{2L1}, s_{2L2}))$  to the list of bilingual suffixes  $S_{SuffixPair}$ .
9:   for each suffix pair  $(s_{iL1}, s_{iL2}) \in S_{SuffixPair}$  do
10:    if  $\exists m, n$  such that  $(ms_{iL1}, ns_{iL2}) \in S_{SuffixPair}$ ,  $u_2 m = u_1$ ,  $v_2 n = v_1$ 
11:      and bilingual stem  $(u_1, v_1)$  and  $(u_2, v_2) \in S_{StemPair}$ , then
12:        replace  $(u_1, v_1)$  by  $(u_2, v_2)$  and  $(s_{iL1}, s_{iL2})$  by  $(ms_{iL1}, ns_{iL2})$  iff
13:         $Strength(s_{iL1}, s_{iL2})$  or  $Strength(m, n) > Strength(ms_{iL1}, ns_{iL2})$ .
14:   for each stem pair  $(p_{iL1}, p_{iL2}) \in S_{StemPair}$ , where  $((p_{iL1} s_{iL1}), (p_{iL2} s_{iL2})) = (w_{iL1}, w_{iL2}) \in L$  do
15:     if  $(s_{iL1}, s_{iL2})$  is not in the list of bilingual suffixes
16:       associated with the bilingual stem  $(p_{iL1}, p_{iL2})$  then
17:         append  $(s_{iL1}, s_{iL2})$  to the suffix list associated with  $(p_{iL1}, p_{iL2})$ .
18:   Cluster the stem pairs sharing similar suffix transformations into bilingual suffix classes.
19: end procedure

```

---

**3.2 Classification**

In this section, we discuss the use of SVM based linear classifier<sup>6</sup> (Fan et al., 2008) in predicting if a given segmentation option corresponds to a valid boundary or not.

$$(p_{1L1}, p_{1L2})(s_{1L1}, s_{1L2}), (p_{2L1}, p_{2L2})(s_{2L1}, s_{2L2}), \dots, (p_{nL1}, p_{nL2})(s_{nL1}, s_{nL2}) \quad (1)$$

In Equation 1, all possible bilingual stems and suffixes associated with a given bilingual word pair  $(w_{iL1}, w_{iL2})$  are represented, where  $(p_{iL1}, p_{iL2})(s_{iL1}, s_{iL2})$  represents a candidate for the bilingual stem and suffix (a possible segmentation boundary). The principle of classification involves learning a function, to infer a binary decision for each split, given all possible segmentations comprising of bilingual stems and suffixes for any given unseen translation.

Each of the possible segmentations (constituting bilingual stem and bilingual suffixes) is a data instance, represented as a feature vector and a target value indicating if the corresponding segmentation is valid (+1), invalid (-1) or unknown (0). We train a binary classifier using the features identified from the training dataset made of the bilingual lexicon and the clusters (bilingual suffix classes) identified during the learning phase. Segmentation boundaries identified for each of the bilingual pairs during the learning phase represent positive samples and all

other possible segmentation options for the bilingual pair represent negative samples. Given all possible splits for a new bilingual pair, the estimated model should predict if each of the candidate segmentations represents a valid boundary (+1) or not (-1).

**3.2.1 Lexicon as Training Data**

The measures discussed below, used in segmenting words into substituent morphemes, are adopted in bilingual framework and are used to derive features to minimally supervise the segmentation.

**Stand-alone Bilingual Pair** We use a binary valued feature indicating if each candidate bilingual stem appears as a stand-alone translation in the lexicon with respect to the candidate segmentation boundary. This knowledge is frequently used in several word-based models and in one of the best performing approaches selected by Hafer *et al.* (Hafer and Weiss, 1974). Instances of bilingual stems appearing as stand-alone bilingual pairs in the lexicon are ‘mountain’  $\Leftrightarrow$  ‘pahAD’ and ‘region’  $\Leftrightarrow$  ‘kShetr’.

**Candidate Boundary Offset (BO)** A pair of index numbers indicating the position of the candidate boundary relative to the beginning and end of the bilingual pair characterises the boundary points. Single-character suffixes, or generally short suffixes are often observed to be spurious than the long ones (Goldsmith, 2001). Index values have been used as multipliers in the function reflecting optimal split

<sup>6</sup><http://www.csie.ntu.edu.tw/~cjlin/papers/liblinear.pdf>

position to deal with the disparity with respect to the frequency of shorter stems and suffixes vs longer ones (Patel et al., 2010). Further, the index values have been used as features in correcting the problem with predecessor variety values resulting from normalisation (Çöltekin, 2010). This knowledge is represented by 4 additional features:

- A pair of integer-valued features corresponding to the offsets from the beginning of the bilingual pair (with respect to candidate boundary). For the bilingual pair, ‘boys’  $\Leftrightarrow$  ‘laDakoM’, with a candidate bilingual stem ‘boy’  $\Leftrightarrow$  ‘laDak’, the offsets<sup>7</sup> are 3 and 3 EN and HI characters, respectively.
- A pair of integer-valued features corresponding to the offsets from the end of the bilingual pair (with respect to candidate boundary). For the example above, the offsets are 1 and 1 EN and HI character, respectively.

**Normalised Successor Entropy (NSE)** The successor entropy is calculated for each stem pair as :

$$H(p_{L1}, p_{L2}) = - \sum_{(s_{L1}, s_{L2}) \in succ(p_{L1}, p_{L2})} \frac{f(p_{L1}s_{L1}, p_{L2}s_{L2})}{f(p_{L1}, p_{L2})} \log_2 \frac{f(p_{L1}s_{L1}, p_{L2}s_{L2})}{f(p_{L1}, p_{L2})} \quad (2)$$

where,  $(p_{L1}s_{L1}, p_{L2}s_{L2})$  is the bilingual string that is formed by concatenation of  $s_{L1}$  to  $p_{L1}$  and  $s_{L2}$  to  $p_{L2}$ ,  $f()$  represents the frequency of the bilingual pairs starting with the given bilingual stem (prefix pair), and  $succ()$  returns all bilingual suffixes (suffix pairs) for the given bilingual stem  $(p_{L1}, p_{L2})$ .

NSE for a candidate stem pair is obtained by dividing the calculated entropy value by the expected value (considering bilingual stems having same length as the candidate stem pair) corresponding to the split position.

**Normalized Predecessor Entropy (NPE)** NPE for a candidate suffix pair is obtained by dividing the calculated predecessor entropy (PE) value by the expected value (considering the bilingual suffixes having same length as the candidate suffix pair) with respect to the split position. PE can be obtained using the Equation 2 by replacing successor with predecessor and switching the concatenation order.

<sup>7</sup>Transcription of HI characters to Latin ones is not character number conservative. But as we work with both character types, offsets must obey the character set in question.

**Normalized Successor Variety (NSV) and Normalized Predecessor Variety (NPV)** We define successor variety as the number of distinct bilingual suffixes that follow a candidate bilingual stem. This count is calculated for each candidate bilingual stem in the training data set. The SV segmentation measure initially proposed by Harris (1970) is employed in numerous word-segmentation tasks (Déjean, 1998; R. et al., 2005; Stein and Potthast, 2007; Bordag, 2008). Further, researches show how this measure could be utilised in improving the segmentation results (Hafer and Weiss, 1974; Çöltekin, 2010).

The variety values are normalised by dividing the calculated value by the expected value (based on the equi-lengthed bilingual stems) with respect to the split position. The NPV value for a candidate bilingual suffix may be calculated similarly. Çöltekin (2010) provide an elaborate analysis of the problems concerning SV values and the suggested improvements using normalized SV scores.

**Bilingual Morpheme Frequency (BMF)** This measure quantifies a candidate bilingual morpheme by the number of distinct translations to which it attaches in the bilingual lexicon.

$$bmf(m_{L1}, m_{L2}) = \text{Number of unique bilingualpairs}(m_{L1}, m_{L2}) \text{ attaches to.} \quad (3)$$

where  $(m_{L1}, m_{L2})$  is the candidate bilingual morpheme (a bilingual stem or a bilingual suffix). This adds 2 features, corresponding to each candidate bilingual stem and the candidate bilingual suffix.

**Generative Strength (GS)** Instead of placing same weight on each bilingual pair when scoring a morpheme, each bilingual pair might be assigned weight based on its generative strength (Dasgupta and Ng, 2007). The generative strength of a bilingual pair is estimated by calculating how many distinct induced bilingual morphemes attach to that bilingual pair. The score of a bilingual morpheme is defined to be the sum of the strengths of the bilingual pairs to which it attaches.

$$gs(m_{L1}, m_{L2}) = \sum_{(w_{iL1}, w_{iL2})} \text{Strength}(w_{iL1}, w_{iL2}). \quad (4)$$

where  $(w_{iL1}, w_{iL2})$  represents the bilingual pair to which the candidate bilingual morpheme  $(m_{L1}, m_{L2})$  attaches. The heuristic has been used in various word-based segmentation tasks to select from among multiple suffixes while stemming a

word form (Pandey and Siddiqui, 2008; Zeman, 2008).

Table 4 (columns 3 and 4) shows the scores for frequent bilingual suffixes using each of the above mentioned scoring functions.

### 3.2.2 Clusters as Training Data

The clusters (bilingual suffix classes) generated in the learning phase is additionally used as training data to model the bilingual suffixes for classification.

#### Cluster-based Bilingual Suffix Length (CBSL)

This is calculated as the number of times a bilingual pair which is  $(l_1, l_2)$  characters contains an  $(sl_1, sl_2)$  character long bilingual suffix, normalized by the total number of bilingual pairs with length  $(l_1, l_2)$  (Brychcín and Konopík, 2015).

#### Cluster-based Bilingual Suffix Probability (CBSP)

This represents the probability that a candidate bilingual morphological extension is a correct bilingual suffix. The clusters generated in learning phase are used to estimate this and is calculated as the number of times the bilingual suffix  $(s_{iL1}, s_{iL2})$  follows the bilingual stem of a translation  $(w_{iL1}, w_{iL2})$  (for each bilingual pair in each cluster), divided by the number of all times  $(w_{iL1}, w_{iL2})$  ends with  $(s_{iL1}, s_{iL2})$  (Brychcín and Konopík, 2015).

### 3.3 Suffix Class Determination and Translation Generation

Given a new translation, upon identification of the segmentation boundary (after classification), we need to identify to which bilingual suffix class the surfaced bilingual suffix and hence the translation belongs. Depending on the bilingual suffix and the stem identified for the given translation, the bilingual pair is classified into one of the bilingual suffix classes identified in the training phase. This is approached as a multi-label classification problem.

SVM based tool namely LIBSVM<sup>8</sup> was used to learn the multi-label classifier. A class is represented as a set of features represented by a feature-value pair and a label. The features are bilingual suffixes that are representatives of a class. For any class, the value in a feature-value pair simply indicates whether the bilingual suffix is a representative of that class (if so, 1) or not (if not, 0).

<sup>8</sup>A library for SVMs - Software available at <http://www.csie.ntu.edu.tw/~cjlin/libsvm>

After the bilingual suffix class for a translation is determined based on the split, new translations are suggested by applying the suffix replacement rules to the identified bilingual stem. For example, given a new bilingual pair *dilemmas* ⇔ *duvidhAein* (Figure 1), the bilingual suffix resulting from segmentation is ('s', 'Aein'). As ('s', 'Aein') is classified as belonging to the bilingual suffix class ('', 'A'), ('s', 'Aein'), the new translation is generated by replacing 's' with '' and 'Aein' with 'A', giving rise to the new bilingual variant *dilemma* ⇔ *duvidhA*.

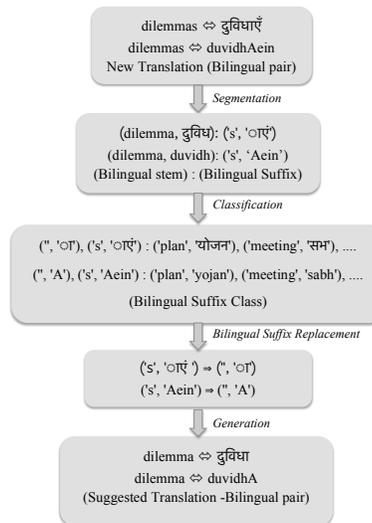


Figure 1: Sample generation

### 3.4 Longest Bilingual Suffix Match (LBSM)

The LBSM technique is used as baseline for identifying bilingual suffixes. After the learning phase, we have different sets of bilingual stems that have been grouped according to their bilingual inflectional classes. We call such sets as Bilingual Suffix Classes. For each translation in the test set, we wish to determine their bilingual inflections (suffixes) and the associated bilingual suffix class. As baseline, we classify each new (unseen) translation in the test set into the class of longest matching bilingual suffix from the bilingual suffix list. For instance, the longest bilingual suffix matching the bilingual pair *conservative* ⇔ *rakshAtmak* is *ative* ⇔ *Atmak* yielding the bilingual stem *conserv* ⇔ *raksh*.

## 4 Experimental Results and Discussion

### 4.1 Data set

We used bilingual pairs taken from EN-HI bilingual lexicon representing single-word translations as the

training data set. Approximately 90% of the entries in the lexicon were acquired from the dictionary<sup>9</sup>. The remaining (10%) entries were partly compiled manually and partially using the Symmetric Conditional Probability based statistical measure from the aligned parallel corpora<sup>10</sup> (Da Silva and Lopes, 1999). The details are as shown in the Table 3.

Description	Total	Training	Test
Bilingual Pairs	58,048	52K	6K
Minimum Length (EN-HI)	3, 3		
Maximum Length (EN-HI)	18, 10		

Table 3: Statistics of the Data set

## 4.2 Bilingual Learning and Generation

The bilingual suffixes (frequently undergoing transformations) recognised using the approach discussed in Section 3.1 are shown in Table 4. Table 5<sup>11</sup> presents the bilingual suffix transformation rules which enable one translation form to be obtained using the other. The grouping in row 1 implies that replacing the suffix ‘s’ with ‘A’ and the suffix ‘oM’ with ‘A’ in the bilingual pair ‘boys’  $\Leftrightarrow$  ‘laDakoM’, yields its bilingual variant ‘boy’  $\Leftrightarrow$  ‘laDaka’.

Bilingual Suffixes	Bilingual Suffixes (Hindi Suffixes transliterated)	Frequency (bmf)	Generative Strength (gs)
(‘, ‘oM)	(‘, ‘A)	10,743	11,240
(‘, ‘oM)	(‘, ‘A)	29,529	30,635
(‘ion’, ‘oM)	(‘ion’, ‘A)	457	567
(‘er’, ‘oM)	(‘er’, ‘A)	428	515
(‘ity’, ‘oM)	(‘ity’, ‘A)	286	340

Table 4: Bilingual Suffixes with frequent replacements

To avoid over-segmentation, we perform the suffix containment check, looking for one candidate bilingual suffix enclosed within another. A true compound bilingual suffix (a combination of multiple candidate bilingual suffixes) is retained based on the observation that the strength of a compound bilingual suffix is less than the strengths of the bilingual suffixes composing it (Dasgupta and Ng, 2007).

**Evaluation** A few of the induced bilingual suffix class based morphological patterns are incomplete as not all the translation forms are seen in the lexicon. Further, distinct surface translation forms due

<sup>9</sup><http://sanskritdocuments.org/hindi/dict/eng-hin`unic.html>, [www.dicts.info](http://www.dicts.info), [hindilearner.com](http://hindilearner.com)

<sup>10</sup>EMILLE Corpus - <http://www.emille.lancs.ac.uk/>

<sup>11</sup>\*Number of times a bilingual suffix co-occurs with another bilingual suffix in the input lexicon (Mahesh et al., 2015)

Bilingual Suffixes	Suffix pair Co-occurrence Score*	Bilingual Stems	
(‘, ‘oM), (‘s’, ‘oM)	27	(‘plant’, ‘पौध’) (‘plant’, ‘pauDh’)	(‘boy’, ‘लड़क’) (‘boy’, ‘laDak’)
(‘, ‘oM), (‘s’, ‘oM)	27	(‘job’, ‘नौकर’) (‘job’, ‘naukar’)	(‘archer’, ‘धनुषधार’) (‘archer’, ‘dhanuShadhaar’)
(‘s’, ‘oM), (‘ous’, ‘oM)	8	(‘mountain’, ‘पर्वत’) (‘mountain’, ‘parvat’)	(‘mountain’, ‘पहाड’) (‘mountain’, ‘pahAD’)
(‘, ‘oM), (‘s’, ‘oM)	3	(‘plan’, ‘योजन’) (‘plan’, ‘yojan’)	(‘meeting’, ‘सभ’) (‘meeting’, ‘saB’)

Table 5: Highly (top 2), less (bottom 2) frequent bilingual suffix replacement rules

to inflection classes result in distinct bilingual suffix classes some of which should be collapsed.

We evaluate the bilingual segments and clustering results indirectly by examining the applicability of induced segments in generating new translations. We first complete the translation lexicon with missing bilingual pairs using bilingual stems and bilingual suffixes learnt using the known bilingual pairs. Generation of missing translation is purely concatenative and is done using the translations in the training data for the chosen bilingual suffix classes (Mahesh et al., 2014). The generated translations are then evaluated. Table 6 shows the results of the learning phase. We calculate the precision for generated translations as the fraction of correctly generated bilingual pairs to total number of bilingual pairs generated. In completing the translation lexicon for missing forms, when both bilingual stems and bilingual suffixes are known, the precision achieved for translation generation reaches 86.52% when compared to the precision of 81.31% obtained using the bilingual learning approach (Mahesh et al., 2014).

Learning Approach	Unique Bilingual Stem Count	Unique Bilingual Suffix Count	Number of Clusters	Generation Precision
IDA2014 (Kavitha et al., 2014)	12,603	781	224	81.31
Proposed-Phase 1	10,224	426	143	86.52

Table 6: Clustering statistics

Table 7<sup>12</sup> shows suggested translation examples. We categorise the generated translations into 3 classes (separated by thick border) based on the degree of correctness. First 3 rows represent acceptable translations (Accept). The following row shows

<sup>12</sup>Two bilingual suffixes are shown per class, though they range from 2 to 5

translation errors (Reject) and the last row represents an inadequate translation (Inadequate). Mentioned errors are briefly explained below:

**Inadequate:** The bilingual pair ‘Russians’ ⇔ ‘*rUsiyOM*’ (last row of the Table 7) is inadequate, as in actual usage, both the singular and plural variants ‘*Russian*’ and ‘*Russians*’ are translated as ‘*rUsI*’. An alternate correct translation would be ‘*rUs vAsI*’.

Generated Translations	Existing Lexicon Entry	Rule used
cleverly ⇔ निपुणता	cleverness ⇔ निपुणता	(‘ly’, ‘ता’), (‘ness’, ‘ता’)
capitalist ⇔ पूंजीवादी materialist ⇔ भौतिकवादी	capitalism ⇔ पूंजीवाद materialism ⇔ भौतिकवाद	(‘ism’, ‘वाद’), (‘ist’, ‘वादी’)
framework ⇔ ढांचा	frameworks ⇔ ढांचे	(‘’, ‘ा’), (‘s’, ‘े’)
world ⇔ लौक (lauk) weeks ⇔ सप्ताहों (sAptahoM)	worldly ⇔ लौकिक (laukik) weekly ⇔ सप्ताहिक (sAptahik)	(‘ly’, ‘िक’), (‘s’, ‘ोँ’)
Russians ⇔ रूसियों (rUsiyOM)	Russian ⇔ रूसी (rUsI)	(‘ian’, ‘ी’), (‘ians’, ‘ियों’)

Table 7: Generated Translations

**Reject:** Incorrect generations are a result of incorrect generalisations. Typical errors correspond to irregular translation forms, specifically, the stem changes before suffixation and misclassifications due to insufficient translation forms. An example for the former class of errors is the generated translation ‘world’ ⇔ ‘lauk’ (row 5), as the correct translated form should be ‘world’ ⇔ ‘lok’. The surface variant ‘worldly’ ⇔ ‘laukik’ is obtained from the stem pair ‘world’ ⇔ ‘lok’ by appending ‘ly’ ⇔ ‘ik’ at the end of the word pair ‘world’ ⇔ ‘lok’. Further, the stem undergoes a change from ‘o’ to ‘au’.

Our approach being purely bilingual suffixation based, does not handle irregular forms and does not capture stem changes prior suffixation.

### 4.3 Minimally supervised learning

The results of segmentation by classification were indirectly evaluated by examining what the induced bilingual segments is expected to facilitate, specifically, in suggesting or generating new translations. In evaluating the generated translations, the Precision (P), Recall (R) and F-measure ( $F_m$ ) are computed as given below:

$$P = t_p / (t_p + f_p), R = t_p / (t_p + f_n), F_m = 2 * P * R / (P + R) \quad (5)$$

where,  $t_p$  denotes the number of times the generated translations were correct,  $f_p$  denotes the number of times the generated translations were incorrect and  $f_n$  denotes the number of times a possible correct translation suggestion was missed. The results for

various features are shown in Table 8. When new translations are given as inputs, the best f-measure of 70.88% is achieved.

Features	Precision	Recall	F-measure
Longest Bilingual Suffix Match	74.41	47.32	57.85
NSV + NPV + BO + Stand-alone pair	75.23	52.54	61.87
NPE + NSE + BO + Stand-alone pair	70.14	57.22	63.02
BMF + GS + CBSP + CBSL + BO + Stand-alone pair	76.21	66.24	70.88

Table 8: Results of minimally supervised learning

## 5 Conclusion and Future Work

We have discussed a minimally supervised approach for learning bilingual segments. The training data prepared using the bilingual learning approach partially serves as the basis for segmentation along with the bilingual lexicon (Mahesh et al., 2014). Various measures used in word segmentation tasks are used as features to represent a boundary/non-boundary condition in a bilingual framework. The segmentation boundary identified for a bilingual pair during the learning phase represent a positive sample and all other possible segmentation options for the bilingual pair represent negative samples. Experiments with distant language pairs and limited training data show that knowing both bilingual stems and bilingual suffixes, missing forms could be generated with the precision of 86.52%. For new translations, the precision falls by 10%.

As future work, direct evaluations should be done by comparing the learned bilingual segments and suffix classes to those in the grammar descriptions for the language pairs under consideration. Learning from bigram equivalents to predict translations for verb forms shall be addressed in the future work.

## Acknowledgements

K. M. Kavitha and Luís Gomes acknowledge the Research Fellowship by FCT/MCTES with Ref. nos., SFRH/BD/64371/2009 and SFRH/BD/65059/2009, respectively, the funded research project ISTRION (Ref. PTDC/EIA-EIA/114521/2009) that provided other means for the research carried out. The authors thank NOVA LINC3, FCT/UNL for providing partial financial assistance to participate in PACLIC 2015, and ISTRION BOX - Translation & Revision, Lda., for providing the valuable consultation.

## References

- Stefan Bordag. 2008. Unsupervised and knowledge-free morpheme segmentation and analysis. In *Advances in Multilingual and Multimodal Information Retrieval*, pages 881–891. Springer.
- Tomáš Brychcín and Miloslav Konopík. 2015. HPS: High precision stemmer. *Information Processing & Management*, 51(1):68–91.
- Çağrı Çöltekin. 2010. Improving successor variety for morphological segmentation. *LOT Occasional Series*, 16:13–28.
- Mathias Creutz and Krista Lagus. 2005. *Unsupervised morpheme segmentation and morphology induction from text corpora using Morfessor 1.0*. Helsinki University of Technology.
- Joaquim Ferreira Da Silva and Gabriel Pereira Lopes. 1999. Extracting multiword terms from document collections. In *Proceedings of the VExTAL: Venezia per il Trattamento Automatico delle Lingue*, pages 22–24.
- Sajib Dasgupta and Vincent Ng. 2007. Unsupervised word segmentation for bangla. *Proceedings of ICON*, pages 15–24.
- Hervé Déjean. 1998. Morphemes as necessary concept for structures discovery from untagged corpora. In *Proceedings of the Joint Conferences on New Methods in Language Processing and Computational Natural Language Learning*, pages 295–298. Association for Computational Linguistics.
- Rong-En Fan, Kai-Wei Chang, Cho-Jui Hsieh, Xiang-Rui Wang, and Chih-Jen Lin. 2008. Liblinear: A library for large linear classification. *The Journal of Machine Learning Research*, 9:1871–1874.
- John Goldsmith. 2001. Unsupervised learning of the morphology of a natural language. *Computational linguistics*, 27(2):153–198.
- Margaret A Hafer and Stephen F Weiss. 1974. Word segmentation by letter successor varieties. *Information storage and retrieval*, 10(11):371–385.
- Harald Hammarström and Lars Borin. 2011. Unsupervised learning of morphology. *Computational Linguistics*, 37(2):309–350.
- Harald Hammarström. 2009. *Unsupervised learning of morphology and the languages of the world*. Ph.D. thesis, Chalmers University of Technology and Göteborg, Gothenburg, December.
- Zellig S Harris. 1970. *From phoneme to morpheme*. Springer.
- Amit Kirschenbaum. 2015. To split or not, and if so, where? Theoretical and empirical aspects of unsupervised morphological segmentation. In *Computational Linguistics and Intelligent Text Processing*, volume 9041 of *LNCS*, pages 139–150. Springer.
- Krister Lindén, Miikka Silfverberg, and Tommi Pirinen. 2009. HFST tools for morphology – An efficient open-source package for construction of morphological analyzers. In *State of the Art in Computational Morphology*, volume 41 of *CCIS*, pages 28–47. Springer.
- Kavitha Karimbi Mahesh, Luís Gomes, and José Gabriel P Lopes. 2014. Identification of bilingual segments for translation generation. In *Advances in Intelligent Data Analysis XIII*, volume 8819 of *LNCS*, pages 167–178. Springer.
- Kavitha Karimbi Mahesh, Luís Gomes, and José Gabriel P Lopes. 2015. Learning clusters of bilingual suffixes using bilingual translation lexicon. In *Mining Intelligence and Knowledge Exploration (Accepted)*. Springer.
- Christian Monson, Jaime Carbonell, Alon Lavie, and Lori Levin. 2009. Paramor and morpho challenge 2008. In *Evaluating Systems for Multilingual and Multimodal Information Access*, volume 5706 of *Lecture Notes in Computer Science*, pages 967–974. Springer.
- Karthik Narasimhan, Regina Barzilay, and Tommi Jaakkola. 2015. An unsupervised method for uncovering morphological chains. *ArXiv preprint arXiv:1503.02335*.
- Amaresh Kumar Pandey and Tanveer J Siddiqui. 2008. An unsupervised hindi stemmer with heuristic improvements. In *Proceedings of the second workshop on Analytics for noisy unstructured text data*, pages 99–105. ACM.
- Pratikumar Patel, Kashyap Popat, and Pushpak Bhattacharyya. 2010. Hybrid stemmer for gujarati. In *23rd International Conference on Computational Linguistics*, page 51.
- Hoifung Poon, Colin Cherry, and Kristina Toutanova. 2009. Unsupervised morphological segmentation with log-linear models. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 209–217. Association for Computational Linguistics.
- Al-Shalabi R., Ghassan Kannan, Iyad Hilat, Ahmad Ababneh, and Ahmad Al-Zubi. 2005. Experiments with the successor variety algorithm using the cutoff and entropy methods. *Information Technology Journal*, 4(1):55–62.
- Hisayuki Sasaoka, Kenji Aaraki, Yoshio Momouchi, and Koji Tochinnai. 1997. Prediction method of word for translation of unknown word. In *Proceedings of the IASTED International Conference, Artificial Intelligence and Soft Computing, Banff, Canada*, page 228. Acta Pr.
- Benjamin Snyder and Regina Barzilay. 2008. Unsupervised multilingual learning for morphological segmentation. *ACL-08: HLT*, page 737.

- Benno Stein and Martin Potthast. 2007. Putting successor variety stemming to work. In *Advances in Data Analysis*, pages 367–374. Springer.
- Richard Wicentowski and David Yarowsky. 2002. *Modeling and learning multilingual inflectional morphology in a minimally supervised framework*. Ph.D. thesis, Ph. D. Thesis. Johns Hopkins University, Baltimore, Maryland.
- Daniel Zeman. 2008. Unsupervised acquiring of morphological paradigms from tokenized text. In *Advances in Multilingual and Multimodal Information Retrieval*, pages 892–899. Springer.

# Selecting Contextual Peripheral Information for Answer Presentation: The Need for Pragmatic Models

**Rivindu Perera, Parma Nand**

School of Computer and Mathematical Sciences

Auckland University of Technology

Auckland, New Zealand

{rivindu.perera, parma.nand}@aut.ac.nz

## Abstract

This paper explores the possibility of presenting additional contextual information as a method of answer presentation Question Answering. In particular the paper discusses the result of employing Bag of Words (BoW) and Bag of Concepts (BoC) models to retrieve contextual information from a Linked Data resource, DBpedia. DBpedia provides structured information on wide variety of entities in the form of triples. We utilize the QALD question sets consisting of a 100 instances in the training set and another 100 in the testing set. The questions are categorized into single entity and multiple entity questions based on the number of entities mentioned in the question. The results show that both BoW (syntactic models) and BoC (semantic models) are not capable enough to select contextual information for answer presentation. The results further reveals that pragmatic aspects, in particular, pragmatic intent and pragmatic inference play a crucial role in contextual information selection in the answer presentation.

## 1 Introduction

Answer Presentation is the final step in Question Answering (QA) which focuses on generating an answer which closely resemble with a human provided answer (Perera, 2012b; Perera, 2012a; Perera and Nand, 2014a). There is also a requirement to associate the answer with additional contextual information when presenting the answer.

This paper focus of exploring methods to extract additional contextual information to present with the

extracted factoid answer. We provide a classification if questions based on the type of the answer required and the number of entities that mentioned in the questions. The question classification is illustrated in Fig. 1. Firstly, question can be categorized based on the information need where questions may require a definition as the answer or a factoid answer which is an information unit (Perera, 2012a; Perera and Nand, 2014a). The definitional questions need definitions which include both direct and related background information and there is no need to further expand the answer with contextual information. So far the way factoid questions presentation involved only the answer itself without contextual information. Recently, Mendes and Coheur (2013) argued that even factoid questions need to present additional information. An advantage of presenting contextual information is that answer is justified by the information provided, so that users can conclude that the answer that is acquired by the system is one that they are searching for.

The rest of the paper is structured as follows. Section 2 explores BoW and BoC models to rank contextual information. Section 3 focuses on presenting the experimental framework. Section 5 presents information on related work and we conclude the paper in Section 6.

## 2 Content selection using weighted triples

This section presents models to rank triples focusing on open domain questions as communicative goals. Open domain questions require knowledge from different domains to be aggregated which making it more challenging compared to simply generating a

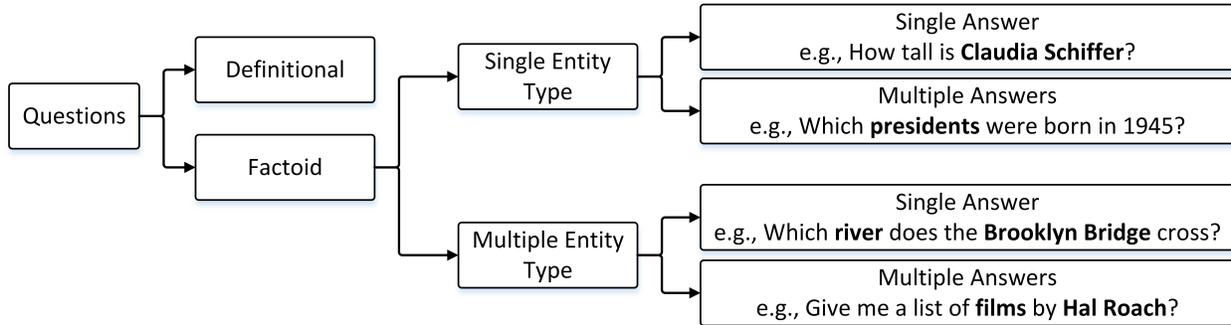


Figure 1: Classification of common question types

content for given single theme topic. Our objective is to select a set of triples which can be used to generate a more informative answer for a given question.

We investigate the problem from two perspectives; as a Bag of Words (BoW) and as a Bag of Concepts (BoC). In the following sections, we discuss the strategy used for ranking and the details on the supporting utilities including domain corpus, reference corpus, triple retrieval, and threshold based selection.

The high level design of the framework used to experiment BoW and BoC is shown in the Fig. 2. The model utilizes two corpora (domain and reference) and selectively used based on the requirement. The domain corpus is constructed using search snippets collected from the web by using information from the question and answer as query terms. The reference corpus represents knowledge about general domain. The model also has utility functions to retrieve triples using SPARQL queries, filter the duplicates, and to perform basic verbalization.

## 2.1 The problem as a Bag of Words

We utilized token similarity, Term Frequency - Inverse Document Frequency (TF-IDF), and Residual Inverse Document Frequency (RIDF) in two flavours which are widely used in information retrieval tasks. The following sections describes these models in detail.

### 2.1.1 Token similarity

Token similarity ranks the triples based on the appearance of the terms in triple and the question being considered. In particular we employ the cosine

similarity (1) to calculate the similarity between the tokenized and stopwords removed question/answer and the triple.

$$\begin{aligned}
 sim_{cosine}(\vec{Q}, \vec{T}) &= \frac{\vec{Q} \cdot \vec{T}}{|\vec{Q}| |\vec{T}|} \\
 &= \frac{\sum_{i=1}^n Q_i T_i}{\sqrt{\sum_{i=1}^n Q_i^2} \sqrt{\sum_{i=1}^n T_i^2}}
 \end{aligned} \tag{1}$$

Here, Q and T represent the question and the triple respectively.

### 2.1.2 Term Frequency – Inverse Document Frequency (TF-IDF)

The TF-IDF (2) is used to rank term ( $t$ ) from the question and answer present in the triple ( $T$ ). A triple is then associated with a weight which is the sum of the weights assigned to the triple terms.

$$\begin{aligned}
 TF - IDF(Q, T) &= \sum_{i \in Q, T} tf_i \cdot idf_i \\
 &= \sum_{i \in Q, T} tf_i \cdot \log_2 \frac{N}{df_i}
 \end{aligned} \tag{2}$$

Where  $tf$  represents the term frequency,  $N$  stands for number of documents in the collection and  $df$  is the number of documents with the corresponding term.  $Q$  represents the question, however in our experiment we tested the possibility of utilizing a domain corpus instead of the original question or the question with the answer.

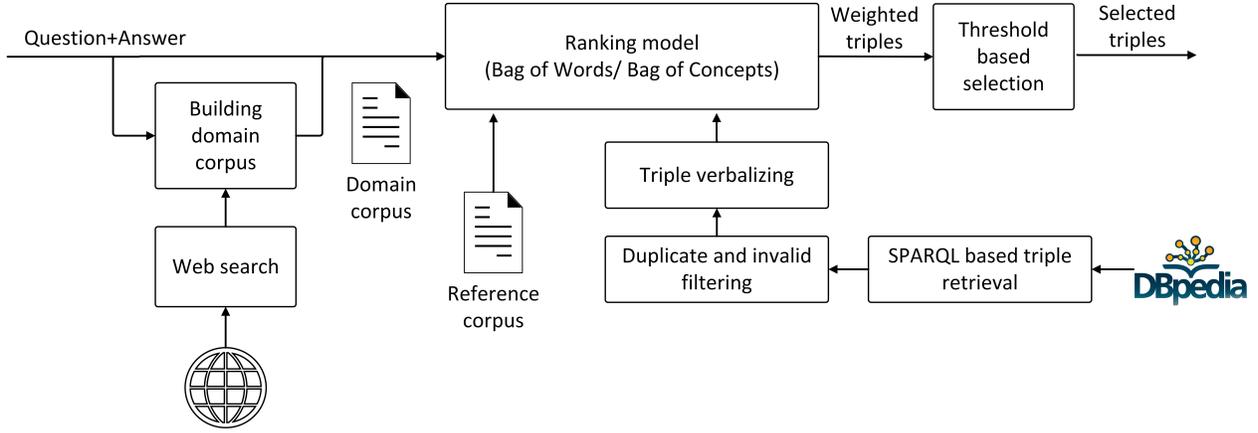


Figure 2: The schematic representation of the content selection framework

### 2.1.3 Okapi BM25

Okapi ranking is an extension to the TF-IDF that is based on the probabilistic retrieval framework.

The Okapi ranking function can be defined as follows:

$$Okapi(Q, T) = \sum_{i \in Q, T} \left[ \log \frac{N}{df_i} \right] \cdot \frac{(k_1 + 1) tf_{i,T}}{k_1 \left( (1 - b) + b \left( \frac{L_T}{L_{ave}} \right) \right) + tf_{i,T}} \cdot \frac{(k_3 + 1) tf_{i,Q}}{k_3 + tf_{i,Q}} \quad (3)$$

Where,  $L_T$  and  $L_{ave}$  represent the length of the triple and average of length of a triple respectively. The Okapi also uses set of parameters where  $b$  is usually set to 0.75 and  $k_1$  and  $k_3$  range between 1.2 and 2.0. The  $k_1$  and  $k_3$  can be determined through optimization or can be set to range within 1.2 and 2.0 in the absence of development data (Manning and Schutze, 1999).

### 2.1.4 Residual Inverse Document Frequency (RIDF)

The idea behind the RIDF is to find content words based on actual IDF and predicted IDF. The widely used methods to IDF prediction is Poisson and K mixture. However, K mixture tends to fit very well with content terms. On the other hand, Poisson deviates from the IDF remarkably and provides non-content words. Given term frequencies in triple collection, predicted IDF can be used to measure the RIDF for a triple as follows:

$$RIDF = \sum_{i \in T} \left( idf_i - \widehat{idf}_i \right) \quad (4)$$

$$= \sum_{i \in T} \left( idf_i - \log \frac{1}{1 - P(0; \lambda_i)} \right)$$

Where  $\lambda_i$  represents the average number of occurrences of term and  $P(0; \lambda_i)$  represents the Poisson prediction of  $df$  where term will not be found in a document. Therefore,  $1 - P(0; \lambda_i)$  can be interpreted as finding at least one term and can be measured using:

$$P(k; \lambda_i) = e^{-\lambda_i} \frac{\lambda_i^k}{k!} \quad (5)$$

Based on the same RIDF concept, we can moderate this to work with term distribution models that fits well with actual  $df$  such as K mixture. The definition of the K-mixture is given below.

$$P(k; \lambda_i) = (1 - \alpha) \delta_{k,0} + \frac{\alpha}{\beta + 1} \left( \frac{\beta}{\beta + 1} \right)^k \quad (6)$$

In K-mixture based RIDF we interpreted the deviation from predicated  $df$  to mark the term as a non-content term.

## 2.2 The problem as a Bag of Concepts

This section explains two BoC models which can rank triples utilizing the semantic representation of the triple collection. In particular, we employ

two widely accepted BoC models; Latent Semantic Analysis (LSA) and adoption of Log Likelihood Distance (LLD) using two corpora. The following sections describe them in detail.

### 2.2.1 Latent Semantic Analysis

This method analysed how triples in the collection can be ranked concept-wise and retrieved related to the question and answer where triples are represented in a semantic space. Such a ranking can expose the original semantic structure of the space and its dimensions (Manning and Schutze, 1999). In particular, we employed the Latent Semantic Indexing (LSI) for each collection of triples associated with the question.

### 2.2.2 Corpus based Log Likelihood Distance (LLD)

The idea behind the implementation of this method is to identify domain specific concepts (compared to the general concepts) and rank triples which contain such concepts. For this we employed a domain corpus (see Section 2.3) and a general reference corpus (see Section 2.4). The model extracts concepts which are related to the domain on the basis of their frequency in domain corpus and general reference corpus. A term that is more frequently seen in a domain corpus compared to the general reference corpus implies that the term is a concept that is used in the domain being considered (Perera and Nand, 2014b; Perera and Nand, 2014c). We utilized the log likelihood distance (He et al., 2006; Gelbukh et al., 2010) to measure the importance as mentioned below:

$$W_t = 2 \times \left( \left( f_t^{dom} \times \log \left( \frac{f_t^{dom}}{f\_exp_t^{dom}} \right) \right) + \left( f_t^{ref} \times \log \left( \frac{f_t^{ref}}{f\_exp_t^{ref}} \right) \right) \right) \quad (7)$$

where,  $f_t^{dom}$  and  $f_t^{ref}$  represent frequency of term ( $t$ ) in domain corpus and reference corpus respectively. Expected frequency of a term ( $t$ ) in domain ( $f\_exp_t^{dom}$ ) and reference corpora ( $f\_exp_t^{ref}$ ) were calculated as follows:

$$f\_exp_t^{dom} = s_{dom} \times \left( \frac{f_t^{dom} + f_t^{ref}}{s_{dom} + s_{ref}} \right) \quad (8)$$

$$f\_exp_t^{ref} = s_{ref} \times \left( \frac{f_t^{dom} + f_t^{ref}}{s_{dom} + s_{ref}} \right) \quad (9)$$

where,  $s_{dom}$  and  $s_{ref}$  represent total number of tokens in domain corpus and reference corpus respectively. Next, we can calculate the weight of a triple ( $\langle$ subject, predicate, object $\rangle$ ) by summing up the weight assigned to each term of the triple

### 2.3 Domain Corpus

The domain corpus is a collection of text related to the domain of the question being considered. However, finding a corpus which belongs to the same domain as the question is challenge on its own. To overcome this, we have utilized an unsupervised domain corpus creation based on a web snippet extraction. The input to this process is a set of extracted key phrases from a question and its answers.

### 2.4 Reference Corpus

The reference corpus is an additional resource utilized for the LLD based contextual information selection. We used the British National Corpus (BNC) as the reference corpus. The selection is influenced by the language used in the DBpedia, British English. However, what is important for the LLD calculation is a term frequency matrix. We have first performed stopword filtering on the BNC and this operation reduced the original size of BNC (100 million words) to 52.3 million words. Next, the term frequency matrix is built using a unigram analysis.

### 2.5 Triple retrieval

The model employs the Jena RDF framework for the triple retrieval. We have implemented a Java library to query and automatically download necessary RDF files from DBpedia.

### 2.6 Threshold based selection

After associating each triple with a calculated weight, we then need to limit the selection based on a particular cut-off point as the threshold ( $\theta$ ). Due

Table 1: Dataset statistics. Invalid questions are those that are already marked by dataset providers as invalid and questions where for which triples cannot be retrieved from DBpedia

	Training	Test
All questions	100	100
Invalid questions	5	10
Single entity questions	47	42
Multiple entity questions	48	48

to the absence of knowledge to measure the  $\theta$  at this stage, it is considered as a factor that needs to be tuned based on experiments. Further discussion on selecting the  $\theta$  can be found in Section 4.

### 3 Experimental framework

#### 3.1 Dataset

We used the QALD-2 training and test datasets, but removed questions which marked as “out of scope” by dataset providers and those for which DBpedia triples did not exist. Table 1 provides the statistics of the dataset, including the distribution of questions in two different question categories, single entity and multiple entity questions.

We have also built a gold triple collection for each question for the purpose of evaluation. These gold triples were selected by analysing community provided answers for the questions in our dataset. Table 1 shows the statistics for both training and testing datasets.

#### 3.2 Results and discussion

The evaluation is carried out using gold triples as described in Section 3.1. The definitions of precision (P), recall (R) and F-score (F\*) are given below:

$$P = \frac{|triples_{selected} \cap triples_{gold}|}{|triples_{selected}|} \quad (10)$$

$$R = \frac{|triples_{selected} \cap triples_{gold}|}{|triples_{gold}|} \quad (11)$$

$$F^* = \frac{2PR}{P + R} \quad (12)$$

The threshold ( $\theta$ ) (measure as a percentage from the total triple collection) value for the ranked triples was experimentally chosen to using the training

Table 2: Statistics related to the gold triple percentage in total triple collection in training dataset. The  $\mu$  represents the mean percentage of gold triples included in the total collection. The  $\sigma$  shows the standard deviation. The *Max%* and *Min%* represent maximum and minimum percentage of gold triples from the total collection respectively.

	$\mu$	$\sigma$	Max%	Min%
Single entity type	68.89	4.28	78.79	63.58
Multiple entity type	30.43	3.88	37.06	22.93

dataset. This threshold value was then used to select triples which were relevant for the testing dataset. The value was experimentally determined by using a combination of precision and recall value from the training data. For an accurate model, the precision is expected to remain constant until it starts selecting the irrelevant triples after which the precision will gradually decrease. Correspondingly, the recall value will increase until the threshold point after which the model will start selecting irrelevant triples, which will start pushing the recall value down. Hence the optimum  $\theta$  value will be the point at the maximum point for both recall and precision which is the maximum score.

Using the  $\theta$  identified from training set, we can then test the model using testing dataset. When measuring the  $\theta$  based on the training dataset it is also important to measure the proportion of gold triples compared to the number of total triples. A set of statistics related to this calculation is shown in Table 2.

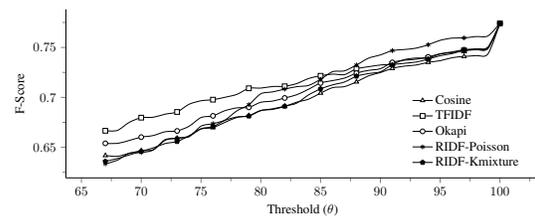


Figure 3: F-score gained by Bag of Words models plotted against threshold for questions with single entity type

According to statistics shown in Table 2 it is clear that the mean percentage of gold triples percentages

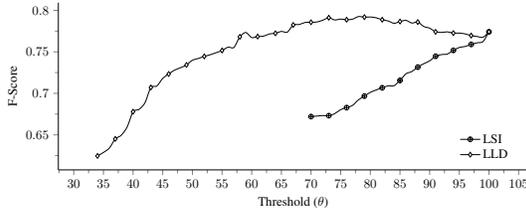


Figure 4: F-score gained by Bag of Concepts models plotted against threshold for questions with single entity type

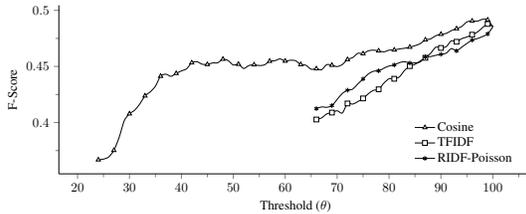


Figure 5: F-score gained by Bag of Words models plotted against threshold for questions with multiple entity types. Okapi and K-mixture based ranking methods completely failed to identify relevant triples.

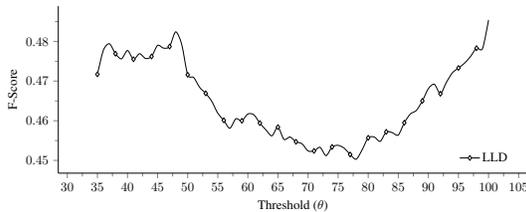


Figure 6: F-score gained by Bag of Concepts models plotted against threshold for questions with multiple entity types. The LSI method failed completely in identifying relevant triples.

are 68.89% for single entity types and 30.43% from the multiple entity types. Furthermore, the maximum and minimum percentages are also near values to the receptive mean values. This encompasses that there is a possibility to find a threshold value for both single and multiple entity types questions. Fig. 3 depicts the evaluation performed on the single entity question category in training dataset using five BoW models under investigation. The results show that the maximum F-score obtained when  $\theta$  is set to 100%. This shows that these models unable to accurately differentiate between the relevant and relevant triples. The BoW models consider only the words

Table 3: Performance of LLD on single entity type question test dataset with 78% threshold

	Precision	Recall	F-score
LLD on Test Dataset (Single Entity)	0.72	0.84	0.76

as features and therefore every entity is assigned the same importance.

The corresponding evaluation performed on the single entity question category in training dataset using BoC models is shown in Fig. 4. Latent Semantic Indexing (LSI) has performed poorly and has not managed to identify a global maximum. However, the Log Likelihood Distance (LLD) has identified a global maximum with a  $\theta$  value of 78%. Furthermore, it has also shown the expected behaviour with an increase in  $\theta$  value. When this threshold value was used to extract triples from the test dataset the results were encouraging. Table 3 shows that LLD has achieved F-score of 0.76 for the testing dataset with a 0.72 precision value. The LLD model outperforms the other models in this context mainly because it also incorporates the domain knowledge (provided through a domain corpus as explained in Section 2.2.2).

Fig. 5 and Fig. 6 depict the evaluation performed on the multiple entity question category from training dataset, for both Bag of Words and Bag of Concepts models. RIDF-Kmixture and Okapi have completely failed without any success in identifying the relevant triples. The Cosine, TF-IDF, and RIDF-Poisson have also not identified the optimum threshold (see Fig. 5). From the BoC models, the LSI method has also failed entirely. The LLD mode has identified a local maximum at  $\theta = 48$ , however the model has not behaved as expected. Furthermore, the global maximum identified at  $\theta = 100$  implies that the model can identify all relevant triples only when the total triple collection is retrieved. This confirms that although a Bag of Concepts model such as LLD performed well in the single entity type questions, none of the models performed well in contextual information selection for multiple entity type questions.

Analysis of the erroneous triples for this experi-

ment revealed that for multiple entity type questions it is important to identify the intent entity from the question. The information from the intent of the question can be used to factor in a weight correction for the triples. This leads to the study the Bag of Narrative (Cambria and White, 2014) model which is essentially based on pragmatic aspects of the language. Section 4 discusses this aspect in detail.

#### 4 Pragmatic aspect in contextual information selection

We introduce two pragmatic based concepts that need to be studied in contextual information selection approaches, derived from psycholinguistics.

- Pragmatic intent (Byram and Hu, 2013) of a question in the perspective of contextual information and,
- Pragmatic inferences (Byram and Hu, 2013; Tomlinson and Bott, 2013) that can be drawn based on already known information.

##### 4.1 Pragmatic Intent

The pragmatic intent of a question in our problem can be defined as the entity that a user is actually intending to know more about. This concept deviates from the two early approaches in query classification; Broder’s taxonomy (Broder, 2002) (classifying queries as informational, transactional, and navigational) and question typologies (Li and Roth, 2006) (determining the answer type for a question).

Consider the examples given in Table 4, where in each question multiple entities are mentioned. The entities are numbered and pragmatic intent is tagged (with code *:i*).

In  $Q_1$ , *Marc Mezvinsky* is the pragmatic intent of the question which is also the expected answer. The same rule applies for  $Q_2$ ,  $Q_3$ , and  $Q_4$ . However, in  $Q_5$  and  $Q_6$  the pragmatic intent is a part of the question (*[MI6]* and *[Natalie Portman]*), but not the answer. This variation makes it difficult to identify the pragmatic intent of a question compared to the question target identification. When presenting contextual information to the user, the information related to the pragmatic intent together with information that is shared by pragmatic intent and other entities need to be given the priority.

Table 5: Example question to illustrate the pragmatic inference used in the information elimination

$Q_7$	Which river does the Brooklyn Bridge in New York cross?
Answer	East River
Triple	(East River, flow through, New York)

##### 4.2 Pragmatic inference

The pragmatic inference is the interpreting information based on the context that it operates on. For example, a storyteller will not mention every incident or fact that happened in a narrative, thus some parts may be left for the reader to interpret using common sense knowledge, open domain knowledge, and knowledge that is already mentioned in the narrative. Applying this well-established psycholinguistic theory in our approach, we noticed several scenarios where we can improve the contextual information by eliminating information that can be pragmatically inferred and prioritizing information that needs for the context.

Consider the question ( $Q_7$ ), answer and the triple provided in Table 5. Using the question and its answer we can infer the following two facts encoded in the triples.

- $F_1$ : Brooklyn Bridge, located in, New York
- $F_2$ : Brooklyn Bridge, crosses, East River

As humans, we can infer that if Brooklyn Bridge is located in New York and if it crosses the East River, then East river must flow through New York, hence it is co-located. Therefore, the triple in Table 5 becomes unimportant for the context because it is already inferred by  $F_1$  and  $F_2$  which can be derived from question and its answer.

The pragmatic inference can also be used to prioritize the information using semantic relations that entities contain. For example consider the two scenarios illustrated in Table 6 where important contextual information can be inferred based on the semantic relationship of the pragmatic intent and entities.

In  $Q_8$  the relation between the entity “Virgin Group” and its co-founders (“Richard Branson” and “Nik Powell”) is in the form of launching a new organization. This makes the information such as cur-

Table 4: Example questions to illustrate the pragmatic intent variation in different questions. Entities are numbered and intent is marked with code *i*

#	Question	Answer
Q <sub>1</sub>	Who is the daughter of [Bill Clinton] <sub>1</sub> married to?	[Marc Mezvinsky] <sub>2</sub> <sup><i>i</i></sup>
Q <sub>2</sub>	Which river does the [Brooklyn Bridge] <sub>1</sub> in [New York] <sub>2</sub> cross?	[East river] <sub>3</sub> <sup><i>i</i></sup>
Q <sub>3</sub>	Which bridge located in [New York] <sub>1</sub> is opened on 19th March 1945?	[Brooklyn Bridge] <sub>2</sub> <sup><i>i</i></sup>
Q <sub>4</sub>	What is the highest place in [Karakoram] <sub>1</sub> ?	[K2] <sub>2</sub> <sup><i>i</i></sup>
Q <sub>5</sub>	In which [UK] <sub>1</sub> city is the headquarters of the [MI6] <sub>2</sub> <sup><i>i</i></sup> ?	[London] <sub>3</sub>
Q <sub>6</sub>	Was [Natalie Portman] <sub>1</sub> <sup><i>i</i></sup> born in the [United States] <sub>2</sub> ?	No

Table 6: Examples illustrate the use of pragmatic inference in information prioritization

#	Question	Answer
Q <sub>8</sub>	Who is the founder of [Virgin Group] <sub>1</sub> ?	[Richard Branson] <sub>2</sub> <sup><i>i</i></sup> and [Nik Powell] <sub>3</sub> <sup><i>i</i></sup>
Q <sub>9</sub>	How often was [Michael Jordan] <sub>1</sub> <sup><i>i</i></sup> divorced?	2

rent positions which are held by its co-founders to be prioritized over other information which is not strongly related to the context of the question. Next, Q<sub>9</sub> is related to Michael Jordan’s marriage. When retrieving contextual information for this question the basic information about his wives such as personal names becomes more important for the context of the question.

## 5 Related work

Benamara and Dizier (2003) present the cooperative question answering approach which generates natural language responses for given questions. In essence, a cooperative QA system moves a few steps further from ordinary question answering systems by providing an explanation of the answer. However, this research lacks the investigation to the information needs of different questions and the process of utilizing cohesive information for the explanation, without redundant text.

Bosma (2005) incorporates the summarization as a method of presenting additional information in QA systems. He coins the term, an *intensive answer* to refer to the answer generated from the system. The process of generating an *intensive answer* is

based on the summarization using rhetorical structures. Several other summarization based methods for QA such as Demner-Fushman and Lin (2006) and Yu et al. (2007) also exist with slightly varying techniques.

Vargas-Vera and Motta (Vargas-Vera and Motta, 2004) present an ontology based QA system, AQUA. Although AQUA is primarily aimed at extracting answers from a given ontology, it also contributes to answer presentation by providing an enriched answer. The AQUA system extracts ontology concepts from the entities mentioned in the question and present those concepts in aggregated natural language.

## 6 Conclusion

This study has examined the role and effectiveness of syntactic and semantic models in contextual information selection for answer presentation. The results showed that the semantic models (e.g., LLD) performed the best for single entity based questions, however the performance dropped for multiple entity questions. An analysis of the multi-entity questions showed that in order to improve performance there is a need to integrate pragmatic aspects into the ranking framework. Further work needs to be done to establish a framework to model pragmatic aspects in contextual information selection. We have already launched the development of the pragmatic framework as discussed in Section 4. Future work will introduce diverse methods of answer presentation in question answering system utilizing contextual information (Perera and Nand, 2015b; Perera et al., 2015; Perera and Nand, 2015a; Perera and Nand, 2015c).

## References

- Farah Benamara and Patrick Saint Dizier. 2003. Dynamic generation of cooperative natural language responses in webcoop. In *9th European Workshop on Natural Language Generation*, Budapest, Hungary. ACL.
- Wauter Bosma. 2005. Extending answers using discourse structure. In *Recent Advances in Natural Language Processing*, Borovets, Bulgaria. Association for Computational Linguistics.
- Andrei Broder. 2002. A Taxonomy of Web Search. *ACM SIGIR Forum*, 36(2):3–10.
- Michael Byram and Adelheid Hu. 2013. *Routledge Encyclopedia of Language Teaching and Learning*. Routledge, Taylor & Francis Group, London, UK.
- Erik Cambria and Bebo White. 2014. Jumping NLP Curves: A Review of Natural Language Processing Research. *IEEE Computational Intelligence Magazine*, 9(2):48–57, May.
- Dina Demner-Fushman and Jimmy Lin. 2006. Answer extraction, semantic clustering, and extractive summarization for clinical question answering. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the ACL - ACL '06*, pages 841–848, Morristown, NJ, USA, July. Association for Computational Linguistics.
- Alexander Gelbukh, Grigori Sidorov, and Liliana Lavilla, Eduardo Chanona-Hernandez. 2010. Automatic Term Extraction Using Log-Likelihood Based Comparison with General Reference Corpus. In *Natural Language Processing and Information Systems*, pages 248–255. Springer Berlin Heidelberg.
- Tingting He, Xiaopeng Zhang, and Ye Xinghuo. 2006. An Approach to Automatically Constructing Domain Ontology. In *20th Pacific Asia Conference on Language, Information and Computation*, pages 150–157, Wuhan. Association for Computational Linguistics.
- Xin Li and Dan Roth. 2006. Learning question classifiers: the role of semantic information. *Natural Language Engineering*, 12(3):229.
- Christopher D. Manning and Hinrich Schütze. 1999. *Foundations of statistical natural language processing*. Massachusetts Institute of Technology.
- Ana Christina Mendes and Luisa Coheur. 2013. When the answer comes into question in question-answering: survey and open issues. *Natural Language Engineering*, 19(01):1–32, January.
- Rivindu Perera and Parma Nand. 2014a. Interaction history based answer formulation for question answering. In *International Conference on Knowledge Engineering and Semantic Web (KESW)*, pages 128–139.
- Rivindu Perera and Parma Nand. 2014b. Real text-cs - corpus based domain independent content selection model. In *IEEE International Conference on Tools with Artificial Intelligence (ICTAI)*, pages 599–606.
- Rivindu Perera and Parma Nand. 2014c. The role of linked data in content selection. In *Pacific Rim International Conference on Artificial Intelligence (PRICAI)*, pages 573–586.
- Rivindu Perera and Parma Nand. 2015a. Generating lexicalization patterns for linked open data. In *Second Workshop on Natural Language Processing and Linked Open Data collocated with 10th Recent Advances in Natural Language Processing (RANLP)*, pages 2–5.
- Rivindu Perera and Parma Nand. 2015b. A multi-strategy approach for lexicalizing linked open data. In *International Conference on Intelligent Text Processing and Computational Linguistics (CICLing)*, pages 348–363.
- Rivindu Perera and Parma Nand. 2015c. Realextext-asg: A model to present answers utilizing the linguistic structure of source question. In *29th Pacific Asia Conference on Language, Information and Computation*.
- Rivindu Perera, Parma Nand, and Gisela Klette. 2015. Realextext-lex: A lexicalization framework for linked open data. In *14th International Semantic Web Conference*.
- Rivindu Perera. 2012a. Ipedagogy: Question answering system based on web information clustering. In *IEEE Fourth International Conference on Technology for Education (T4E)*.
- Rivindu Perera. 2012b. *Scholar: Cognitive Computing Approach for Question Answering*. Honours thesis, University of Westminster.
- John Tomlinson and Lewis Bott. 2013. How intonation constrains pragmatic inference. In *35th Annual Conference of the Cognitive Science Society*, Berlin, Germany. Cognitive Science Society.
- M Vargas-Vera and E Motta. 2004. Aqua-ontology-based question answering system. In *Mexican International Conference on Artificial Intelligence*, Mexico City, Mexico. Springer-Verlag.
- Hong Yu, Minsuk Lee, David Kaufman, John Ely, Jerome A. Osheroff, George Hripcsak, and James Cimino. 2007. Development, implementation, and a cognitive evaluation of a definitional question answering system for physicians. *Journal of Biomedical Informatics*, 40:236–251.

# RealText<sub>asg</sub>: A Model to Present Answers Utilizing the Linguistic Structure of Source Question

**Rivindu Perera, Parma Nand**

School of Computer and Mathematical Sciences

Auckland University of Technology

Auckland, New Zealand

{rivindu.perera, parma.nand}@aut.ac.nz

## Abstract

Recent trends in Question Answering (QA) have led to numerous studies focusing on presenting answers in a form which closely resembles a human generated answer. These studies have used a range of techniques which use the structure of knowledge, generic linguistic structures and template based approaches to construct answers as close as possible to a human generate answer, referred to as human competitive answers. This paper reports the results of an empirical study which uses the linguistic structure of the source question as the basis for a human competitive answer. We propose a typed dependency based approach to generate an answer sentence where linguistic structure of the question is transformed and realized into a sentence containing the answer. We employ the factoid questions from QALD-2 training question set to extract typed dependency patterns based on the root of the parse tree. Using identified patterns we generate a rule set which is used to generate a natural language sentence containing the answer extracted from a knowledge source, realized into a linguistically correct sentence. The evaluation of the approach is performed using QALD-2 testing factoid questions sets with a 78.84% accuracy. The top-10 patterns extracted from training dataset were able to cover 69.19% of test questions.

## 1 Introduction

Question Answering (QA) comprises of four main tasks; question processing, answer search, answer extraction, and answer presentation. The first three

tasks focus on extracting the answer while the last aims to present the extracted answer in a human-like format. With the rise of trend towards building human-competitive QA systems, there been a corresponding demand for the extracted to be presented in a human competitive form rather than the bare answer as a single word or a phrase. A wide range of answer presentation schemes have been reported including user tailored answers (Mendes and Coheur, 2013; Maybury, 2008; Kolomiyets and Moens, 2011; Perera and Nand, 2014a), justification based answers (Mendes and Coheur, 2013; Maybury, 2008; McGuinness, 2004; Saint-Dizier and Moens, 2011), presentation of paragraph level text summaries with the extracted answer (Mendes and Coheur, 2013; Lin et al., 2003; Perera, 2012b; Perera, 2012a), presentation of hot links with answers (McGuinness, 2004), and presentation of navigable related answers and contextual information (Saint-Dizier and Moens, 2011; Perera and Nand, 2015a; Perera and Nand, 2014b; Perera and Nand, 2014c). All of the mentioned models aim to build an answer which closely resembles a human generated answer. However, an approach that has not been explored in the mentioned models is to exploit the structure of the question in the formulation of the answer. A human generated answer is based both on the answer structure as well as how the question was formulated (Singer, 2013). For example, given the question “Which river does the Brooklyn Bride cross?”, the expected answer sentence would be of the form of “The Brooklyn Bridge crosses East River”.

It is essential to understand the types of questions and their linguistic structure in order to suc-

successfully generate a sentence with the answer embedded in it. The questions can be divided in to two main categories based on their interrogative categories; *wh*-interrogative and polar interrogatives. A *wh*-interrogative is aimed at getting an answer which represents another entity or a property of a resource mentioned in the question, on the other hand a polar interrogative requests a true/false (yes/no) answer. These two types require two different answer sentence generation schemes; *wh*-interrogatives require to embed the answer to the modified source question linguistic structure and the polar interrogatives need to transform the same question without further embedding, however it still needs modification based on the answer. Table 1 shows the interrogative types with examples and Part-Of-Speech (POS) tags associated with them and the expected answer sentences.

This paper focuses on answer sentence generation based on typed dependency parsing. To the best of our knowledge, no previous study has investigated this method of generating an answer sentence utilizing the source question’s linguistic structure. The methodology we introduce here is based on linguistics. The core idea is that the generation of an answer sentence is initiated by identifying the root of the parse tree and then proceed to build the sentence using the nominal subject, a key feature of a Subject-Verb-Object (SVO) style language such as English. In order to identify the grammatical relation that holds the parts of question with the root, we employ typed dependency parse of the complete question. The typed dependency based patterns extracted using a training dataset are used to construct the framework. Answer merging and further realization of the sentence are implemented in order to provide a human-like natural language answer sentence. The complete framework (implemented in Java) and the datasets are available for download from the project website<sup>1</sup>.

The remaining part of the paper proceeds as follows. Section 2 introduces the methodology of answer sentence generation. We discuss the process under four main themes; extracting syntactic patterns, applying patterns to new questions, answer merging, and further realization. Section 3 describes

the experimental framework including the results. A discussion on related work which investigates different answer presentation methods in natural language is presented in Section 4. Section 5 concludes the paper with an overview of future work.

## 2 Answer Sentence Generation

In this section we explain the Answer Sentence Generation (ASG) process which has a pipeline architecture as shown in Fig.1. The process is comprised of three main components; pattern processing (pattern extraction and application), answer merging, and sentence realization. The pattern processing component is responsible of deriving typed dependency based patterns to transform a question back into a natural sentence. It is also responsible for identifying and applying the appropriate pattern based on the typed dependency parse of a question. The answer merging module embeds the answer in *wh*-interrogatives preserving the naturalness of the sentence. The sentence realization module applies grammar based realization if needed and further realizes the answer sentence.

### 2.1 Pattern extraction

The pattern identification process first identifies the interrogative type of the question. We employed the Stanford parser<sup>2</sup> to parse the question and to identify the POS tag of a *wh*-determiner. However, POS tagging itself cannot be used to classify questions because of two reasons. Firstly, a sentence can be formed using an embedded interrogative such as “I wonder what he likes to eat for the dinner” or “Do only what is assigned to you”. In aforementioned examples, the former is an embedded interrogative to explain the speaker’s perspective and the latter is a command, however both cannot be considered as questions. The second reason is that when forming a sentence using relative clauses (both restrictive and non-restrictive) the joining token is also POS tagged as *wh*-determiner. For instance, the sentence “Chess is a good game that is interesting too” contains the *wh*-determiner (*WRB*) POS tag (based on Penn Treebank guidelines) associated with token “*that*”. Due to these factors, we consider three features, the POS

<sup>1</sup><http://rivinduperera.com/information/realtext.html>

<sup>2</sup><http://nlp.stanford.edu/software/lex-parser.shtml>

Table 1: Interrogative types with examples and associated POS tags. POS tags are compliant with the Penn Treebank guidelines.

	Wh-interrogative	Polar interrogative
Interrogative tokens	Who, What, Where, Which, When,	Is, Are, Was, Were
POS tags	How WP, WRB ,WDT	VBZ, VBP, VBD
Question - 1	Which river does the Brooklyn Bridge	Was Natalie Portman born in the United
Answer	cross? East River	States? False/No
Answer sentence	The Brooklyn Bridge crosses East	Natalie Portman was not born in the
Question - 2	River. How many films did Hal Roach pro-	United States. Is Cristian Bale starring in Batman Be-
Answer	duce? 509	gins? True/Yes
Answer sentence	Hal Roach produced 509 films.	Cristian Bale is starring in Batman Be-
		gins.

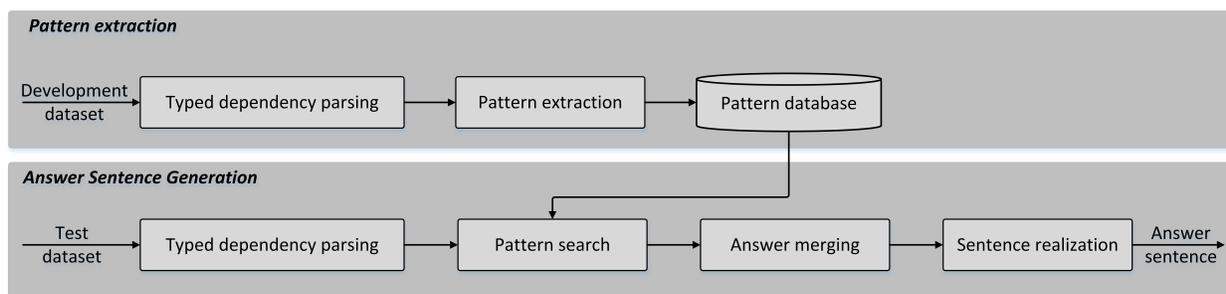


Figure 1: Schematic representation of the overall Answer Sentence Generation (ASG) process

tag, the relative position in the sentence, and whether they belong to the wh-lexicons (who, when, what, etc.).

Once the interrogative type is identified, the patterns can be extracted using a development question set. Each pattern is a collection of first level typed dependency relations as a directed graph based on the root node of the parse tree whose nodes are generic. The order of typed dependency relations are not significant as linguistic structure may vary based on the formation of the question. Table 2 shows some patterns and with example questions.

Each extracted pattern is associated with rule sets to generate a sentence. These rules specify how nominal subject, direct object, clausal complements, and other typed dependency relations should be aggregated to form a natural sentence.

## 2.2 Identifying and applying patterns

Once the pattern database is built, an appropriate pattern for a given question can be retrieved by analysing the typed dependency parse of that question. In the identification process, the order of the relations in the typed dependency pattern is insignificant as we only consider the relations from the root node to another generic node.

When applying the pattern, the parse tree is transformed to a list of phrases based on the root-based relations. Also more importantly, the phrase is generated based on the order of appearance of tokens in the source question. An example scenario of phrase extraction is shown in Table 3. These phrases are then aggregated based on the rules specified for each pattern.

Table 2: Syntactic patterns extracted from Typed dependency relations. The pattern is derived from the typed dependencies from the root token. The sign *X* represents a slot which can be replaced with a single or multiple tokens even if there exist typed dependency relations among those multiple tokens. The sign *R* represents the root token of the parse tree.

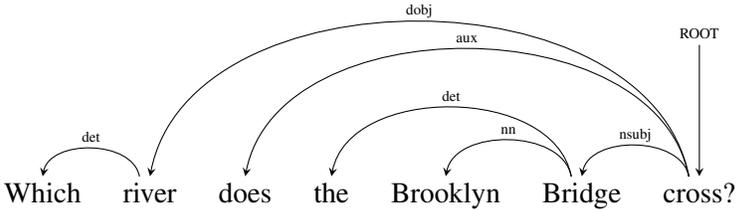
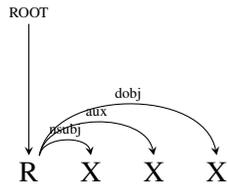
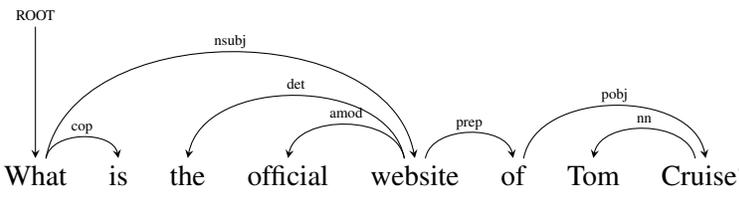
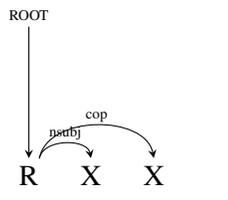
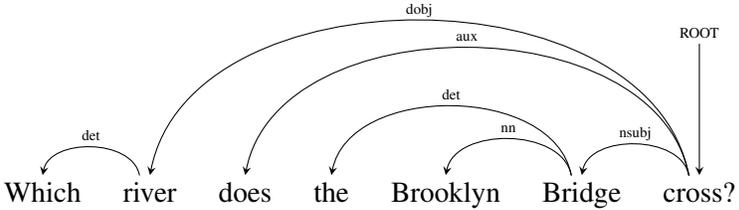
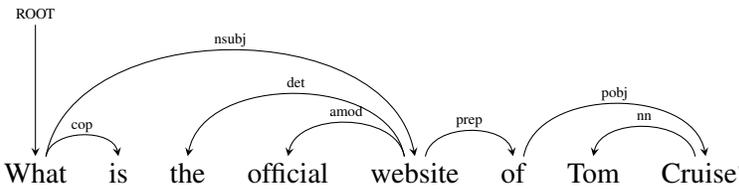
Type dependency	Extracted pattern
	
	

Table 3: Phrase extraction from typed dependencies

Type dependency	Extracted phrases
	<ul style="list-style-type: none"> <li>(i) Which river</li> <li>(ii) does</li> <li>(iii) the Brooklyn Bridge</li> <li>(iv) cross</li> </ul>
	<ul style="list-style-type: none"> <li>(i) What</li> <li>(ii) is</li> <li>(iii) the official website of Tom Cruise</li> </ul>

### 2.3 Answer merging

It is also required to embed the answer to the syntactic structure when the pattern has been identified to transform the question back into natural language sentence. In *wh*-interrogatives this require embedding another language segment, however for polar interrogatives this component should target on modifying the polar token based on the answer.

For *wh*-interrogatives, we have designed the model to embed the answer based on the type of the

*wh*-token. This model is depicted in Table 4 for six different *wh*-tokens. It is also important to note that the *wh*-token “*why*” is not considered, since the current paper focuses only on factoid questions (e.g., how tall is Claudia Schiffer?) and definitional questions (e.g., why the sky is blue?) which often start with *wh*-token “*why*” are out of the current scope of the paper. The main reason for this elimination is that definitional questions does not require answer sentences as answer is a explanation.

Table 4: Answer merging schema for wh-interrogatives. Existing preposition is a one that is already appeared in the phrase and new prepositions are added based on the answer.

Wh-token	Merging schema	Example phrases	Merged answer
Which	Existing preposition + Answer	in which country	in New Zealand
What	Existing preposition + Answer	for What city	for London
Whom	Existing preposition + Answer	for whom	for Barack Obama
How	Naturalized answer (once/ twice/ thrice)	how often	twice
	Answer + Rest of the Phrase	how many films	509 films
When	New preposition + Answer	When	in 1990
Where	New preposition + Answer	Where	in New York

In addition to the answer merging schema shown in Table 4, the model also embeds measurement units and converts numbers to words. For example, if a number has appeared as the first word of an answer sentence, they are converted to a lexical form (e.g., 31  $\Rightarrow$  Thirty one). If a question requires the height of an entity (e.g., person or mountain) as the answer then appropriate measurement unit is added which is extracted from the knowledge source utilized for the answer. However, the query built to extract the answer needs to be analysed in order to determine whether answer requires a measurement unit. The queries generated in QA systems highly depend on the answer extraction source. For example, a QA system which utilizes a database will employ SQL as the query language and transforming the natural language question into a SQL query will be a major task for the query processing module of the QA system. The experiments described in this paper utilizes a Linked Data resource, DBpedia, as the answer extraction data resource (more information on this selection is described in Section 3.1). Use of DBpedia as the data resource required us to implement a SPARQL<sup>3</sup> (SPARQL Protocol and RDF Query Language) processing module. In particular, we used the Jena<sup>4</sup> to parse the SPARQL query and identify queried predicate from the SPARQL. The module then searches the queried predicate in a local lexicon database (this is built as a different task in this research (Perera et al., 2015; Perera and Nand, 2015b; Perera and Nand, 2015c)) to identify whether it is associated with a measurement unit. Listing 1 depicts an example sce-

nario of identifying the measurement unit associated with *height* ontology property of DBpedia.

## 2.4 Sentence realization

The sentence realization is based on a linguistic realization module which can further realize the answer sentence. However, by this stage, the answer sentence is nearly built except for the verb inflections. Therefore, this modules focuses on realization of periphrastic tense in occasions where the verb can be inflected without compromising the semantics (e.g., does cross  $\Rightarrow$  crosses). Also more importantly the formation of active voice based questions (identified using POS tagging) often requires periphrastic tense embedded in the question (e.g., Which river *does* the Brooklyn Bridge *cross*?  $\Rightarrow$  does cross  $\Rightarrow$  crosses). We used a specially built verb information database to identify different inflections of verbs. This database was built using VerbNet and contains 3773 records where each corresponds to a unique verb. However, VerbNet does not provided verb inflections. Therefore, we reverse engineered the Porter stemming algorithm (Porter, 1980) to generate the verb inflections.

An example set of records of this database is shown in Table 5.

## 3 Experimental framework

This section explains the experimental framework used to evaluate the answer sentence generation process. Due to absence of a method that can be directly compared to, we report the experiments in various dimensions; syntactic accuracy, execution time, and memory requirements. The last two factors has been added to the evaluation as QA has now moved closer

<sup>3</sup>SPARQL is the query language used for Linked Data

<sup>4</sup><https://jena.apache.org/>

```

PREFIX res: <http://dbpedia.org/resource/>
PREFIX dbo: <http://dbpedia.org/ontology/>
SELECT DISTINCT ?height
WHERE {
  res:Claudia_Schiffer dbo:height ?height .
}

```

⇒ ?height ⇒ dbo:height ⇒ meters (m)

Listing 1: An example scenario of identifying the measurement unit associated with queried predicate by parsing the SPARQL query

Table 5: An example set of records from verb information database. (Base = Base form of the verb, PT = Past tense, PP = Past participle, TPS = Third person singular, Frames = aggregation of all frames found for the verb in VerbNet)

Base	PT	PP	ING	TPS	Frames
abandon	abandoned	abandoned	abandoning	abandons	NP V NP.initial_location
abase	abased	abased	abasing	abases	NP V NP.patient, NP V ADV-Middle, NP V NP PP.instrument
abash	abashed	abashed	abashing	abases	NP.cause V NP, NP V NP, NP V ADV-Middle, NP V NP-PRO-ARB, NP V NP ADJ

to achieving the long-held illusive goal of accuracy, hence we have now started to also look at real-time performance and computational efficiency.

### 3.1 Datasets

The experiments were carried out using factoid questions extracted from QALD-2 datasets. The QALD-2 training dataset is used to extract typed dependency patterns (as the development dataset) and the testing dataset is used to evaluate the framework.

The statistics related to the dataset is summarized in Table 6. The training set contained a record “Give me the homepage of Forbes”. This record does not form a linguistic representation of any type of interrogative question and therefore could not be considered for pattern extraction.

### 3.2 Results and discussion

The evaluation of the framework focused on two aspects; the syntactic and semantic accuracy of the approach, and the memory and processing requirement. The latter was employed to identify the viability of the methodology for real-time systems.

We were able to identify 18 distinct wh-interrogative patterns and 7 polar interrogative pat-

Table 6: Statistics related to the question sets

	Total	Factoid	Wh-interrogatives	Polar interrogatives
Training set	100	50	41	8
Testing set	99	52	43	9

terns. However, based on the syntactic structure identified, these patterns can be generalized under certain interrogative patterns. Throughout this study we will be referring to the extended list of patterns (18 wh-interrogative and 7 polar interrogative patterns). Using these patterns, answer sentences were generated for the testing dataset with a 78.84% accuracy. Except for 11 questions where the framework completely failed to generate answer sentences, all others were syntactically and semantically accurate. These 11 questions include 5 wh-interrogatives and 6 polar interrogatives. The framework failed to generate answer sentences for these questions mainly due to the absence of rules (for 10 questions) and

the errors in the typed dependency parse (for 1 question). Table 7 shows a selected set of generated answer sentences together with questions.

It is also important to analyse the coverage provided for the test dataset by the extracted patterns from training dataset. Fig. 2 shows the number of testing dataset cases covered by top-k patterns extracted from training dataset. The top-10 patterns were able successfully cover 69.19% of the questions from the testing dataset. Furthermore, the coverage of 51.91% of the questions through top-4 patterns shows that the top patterns are highly representative.

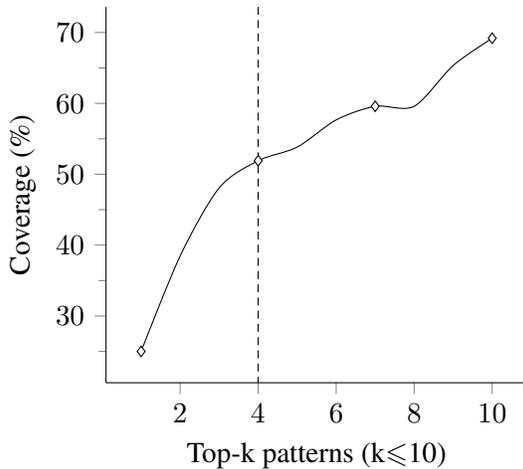
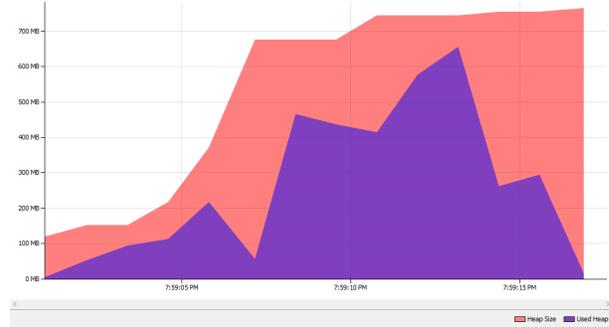


Figure 2: The coverage provided for the test dataset by the extracted patterns from training dataset. The coverage is depicted as a percentage of questions transformed accurately by top-k patterns which are extracted from training dataset. The dashed line shows the point where 50% coverage is reached.

Another aspect of the evaluation of the framework was to analyse the memory and execution analysis. Fig.3(a) and Fig.3(b) depicts the heap memory allocation for the answer sentence generation process and execution time analysis respectively. The remarkably high heap memory allocation (more than 600 Mb) indicates that the current architecture needs further work to work in limited-memory environments. However, the execution time analysis shows that the framework can be adapted to real-time systems.



(a) Heap memory allocation

Typed dependency parsing	2539ms	96.81%
Interrogative type identification	1.82ms	0.06%
Wh-interrogatives	80.1ms	3.04%
Rule identification	2.5ms	0.09%
Rule application	77.6ms	2.95%
Polar interrogatives	1.7ms	0.06%
Rule identification	0.208ms	0.01%
Rule application	1.492ms	0.05%

(b) Execution time for key components

Figure 3: Memory and execution analysis for ASG process. The analysis is performed for test dataset which contains 52 factoid question.

#### 4 Related work

A considerable amount of literature has been published on answer presentation. However, to the best of our knowledge answer sentence generation is not studied in any present model. Therefore, this section provides a broader review of answer presentation.

Benamara and Dizier (Benamara and Dizier, 2003) present the cooperative question answering approach which generates natural language responses for given questions. The idea of cooperative question answering dates back to 1986 with the invention of cooperative interface for information systems. Then several researchers involved the cooperative interfaces for QA systems. In essence, a cooperative QA system moves a few steps further from ordinary question answering systems by providing an explanation of the answer, describing if the system is unable to find an answer or by providing links to the user to get more information for the given question. However, the development of this description is entirely based on the external knowledge source, and the linguistic structure is not considered to present the answer.

A successful attempt to move beyond the ex-

Table 7: Selected set of generated answer sentences together with questions

QALD Id	Question	Answer sentence
2	Who was the successor of of John F. Kennedy?	The successor of John F. Kennedy was Lyndon B. Johnson.
4	How many students does the Free University in Amsterdam have?	The Free University in Amsterdam has 22730 students.
20	How tall is Michael Jordan?	Michael Jordan is 1.9812m tall.
53	What is the ruling party in Lisbon?	The ruling party in Lisbon is Socialist Party (Portugal).
78	Was Margaret Thatcher a chemist?	Margaret Thatcher was a chemist.

act answer by presenting users with additional information in sentence form is presented by Bosma (Bosma, 2005) utilizing summarization techniques. In this research Bosma (Bosma, 2005) assumes that a QA system has already extracted a sentence that contains the exact answer. Then based on this candidate sentence, Bosma (Bosma, 2005) tries to generate an answer response by utilizing information acquired from a collection of sentences. He coins the term, an intensive answer to refer to the answer generated from the system. The process of generating intensive answer is based on summarization using rhetorical structures.

Another answer presentation method based on summarization is presented by Demner-Fushman and Lin (Demner-Fushman and Lin, 2006). They introduce the concept of extractive summaries to present with extracted answer. This model has some similarity with the one presented by Bosma (Bosma, 2005), but like in Bosmas model, Demner-Fushman and Lin (Demner-Fushman and Lin, 2006) do not make use of specifically generated weighted graph to identify relevant sentences to include in the summary. Instead Demner-Fushman and Lin (Demner-Fushman and Lin, 2006) generates the answer by aggregating the top three ranked sentences by a regression model.

Vargas-Vera and Motta (Vargas-Vera and Motta, 2004) present an ontology based QA system, AQUA. Although AQUA is primarily aimed at extracting answers from a given ontology, it also contributes to answer presentation by providing an enriched answer. The AQUA system extracts ontology concepts from the entities mentioned in the question and present those concepts in aggregated natural lan-

guage. However, the research does not contribute towards identifying the most appropriate context for the given question and answer. In addition, no specific effort is taken to identify how the aggregated concepts need to presented. However, the benefit that researchers achieved by building the enriching module on top of an ontology is that the related information can be easily acquired using the relations in the ontology.

## 5 Conclusion and future work

The purpose of the current study was to build a framework to generate answer sentences using the linguistic structure of the question with embedded answer. At the core of the framework, we used typed dependency based patterns extraction and the generated sentence was further realized through multiple strategies. We designed the experiment to determine the accuracy and as well as the resource requirements. The results confirmed that the approach can be successfully applied in QA systems with a reasonable level of accuracy. However, the study found that the framework needs further work to extend it to the mobile platforms as the current architecture consumes considerable memory during execution, mainly due to loading of the pre-trained dependency parser model. In addition, the current study focused only on grammatical accuracy. We plan to extend the evaluation to consider other factors of answer sentences including suitability and cohesion. In addition to aforementioned experimental investigations, further studies need to be carried out to enrich the proposed strategy as well as combine it with other possible strategies to further advance the answer presentation.

## References

- Farah Benamara and Patrick Saint Dizier. 2003. Dynamic Generation of Cooperative Natural Language Responses in WEBCOOP. In *9th European Workshop on Natural Language Generation (ENLG-2003) at EACL 2003*, Budapest, Hungary. Association for Computational Linguistics.
- Wauter Bosma. 2005. Extending answers using discourse structure. In *Recent Advances in Natural Language Processing*, Borovets, Bulgaria. Association for Computational Linguistics.
- Dina Demner-Fushman and Jimmy Lin. 2006. Answer extraction, semantic clustering, and extractive summarization for clinical question answering. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the ACL - ACL '06*, pages 841–848, Morristown, NJ, USA, July. Association for Computational Linguistics.
- Oleksandr Kolomiyets and Marie-Francine Moens. 2011. A survey on question answering technology from an information retrieval perspective. *Information Sciences*, 181(24):5412–5434, December.
- Jimmy Lin, Dennis Quan, Vineet Sinha, Karun Bakshi, David Huynh, Boris Katz, and David R. Karger. 2003. What Makes a Good Answer? The Role of Context in Question Answering. In *Interact*.
- Mark Maybury. 2008. New Directions In Question Answering. In Tomek Strzalkowski and Sanda M. Harabagiu, editors, *Advances in Open Domain Question Answering*, volume 32 of *Text, Speech and Language Technology*, chapter New Direct. Springer Netherlands, Dordrecht.
- D L McGuinness. 2004. Question answering on the semantic Web.
- Ana Christina Mendes and Luisa Coheur. 2013. When the answer comes into question in question-answering: survey and open issues. *Natural Language Engineering*, 19(01):1–32, January.
- Rivindu Perera and Parma Nand. 2014a. Interaction history based answer formulation for question answering. In *International Conference on Knowledge Engineering and Semantic Web (KESW)*, pages 128–139.
- Rivindu Perera and Parma Nand. 2014b. Real text-cs - corpus based domain independent content selection model. In *IEEE International Conference on Tools with Artificial Intelligence (ICTAI)*, pages 599–606.
- Rivindu Perera and Parma Nand. 2014c. The role of linked data in content selection. In *Pacific Rim International Conference on Artificial Intelligence (PRICAI)*, pages 573–586.
- Rivindu Perera and Parma Nand. 2015a. Answer presentation with contextual information: A case study using syntactic and semantic models. In *28th Australasian Joint Conference on Artificial Intelligence*.
- Rivindu Perera and Parma Nand. 2015b. Generating lexicalization patterns for linked open data. In *Second Workshop on Natural Language Processing and Linked Open Data collocated with 10th Recent Advances in Natural Language Processing (RANLP)*, pages 2–5.
- Rivindu Perera and Parma Nand. 2015c. A multi-strategy approach for lexicalizing linked open data. In *International Conference on Intelligent Text Processing and Computational Linguistics (CICLing)*, pages 348–363.
- Rivindu Perera, Parma Nand, and Gisela Klette. 2015. Realex: A lexicalization framework for linked open data. In *14th International Semantic Web Conference*.
- Rivindu Perera. 2012a. Ipedagogy: Question answering system based on web information clustering. In *IEEE Fourth International Conference on Technology for Education (T4E)*.
- Rivindu Perera. 2012b. *Scholar: Cognitive Computing Approach for Question Answering*. Honours thesis, University of Westminster.
- M.F. Porter. 1980. An algorithm for suffix stripping. *Program*, 14(3):130–137.
- Patrick Saint-Dizier and Marie Francine Moens. 2011. Knowledge and reasoning for question answering: Research perspectives. *Information Processing and Management*, 47:899–906.
- Murray Singer. 2013. *Psychology of Language: An Introduction to Sentence and Discourse Processes*. Psychology Press.
- M Vargas-Vera and E Motta. 2004. AQUAontology-based question answering system. In *Mexican International Conference on Artificial Intelligence*, Mexico City, Mexico. Springer-Verlag.

# Learning under Covariate Shift for Domain Adaptation for Word Sense Disambiguation

**Hiroyuki Shinnou, Minoru Sasaki, Kanako Komiya**

Ibaraki University, Department of Computer and Information Sciences

4-12-1 Nakanarusawa, Hitachi, Ibaraki JAPAN 316-8511

hiroyuki.shinnou.0828@vc.ibaraki.ac.jp,

{minoru.sasaki.01, kanako.komiya.nlp}@vc.ibaraki.ac.jp

## Abstract

We show that domain adaptation for word sense disambiguation (WSD) satisfies the assumption of covariate shift, and then solve it by learning under covariate shift. Learning under covariate shift has two key points: (1) calculation of the weight of an instance and (2) weighted learning. For the first point, we employ unconstrained least squares importance fitting (uLSIF), which models the probability density ratio of the source domain against a target domain directly. Additionally, we propose weight only to the particular instance and using a linear kernel rather than a Gaussian kernel in uLSIF. For the second point, we employ a support vector machine (SVM) rather than the maximum entropy method (ME) that is commonly employed in weighted learning. Three corpora in the Balanced Corpus of Contemporary Written Japanese (BCCWJ) and 16 target words were used in our experiment. The experimental results show that the proposed method demonstrates the highest average precision.

## 1 Introduction

Supervised learning methods have been used in many natural language processing tasks. In supervised learning, we create training data for the target task from corpus A and learn a classifier from the training data. This classifier performs well for test data in corpus A; however, it does not perform well for test data in corpus B, which is different from cor-

pus A. This is the problem of domain adaptation<sup>1</sup>. In this paper, we deal with domain adaptation for word sense disambiguation (WSD).

WSD identifies the sense  $c \in C$  of an ambiguous word  $w$  in a sentence  $\mathbf{x}$ . This problem can be solved by the following equation:

$$\arg \max_{c \in C} P(c|\mathbf{x}).$$

The above equation can be solved using supervised learning. However, the domain adaptation problem occurs in a real task. In domain adaptation,  $P_s(c|\mathbf{x})$  can be derived from source domain  $S$ ; therefore, we must estimate  $P_t(c|\mathbf{x})$  in the target domain  $T$  using  $P_s(c|\mathbf{x})$  and other data. Note that the sense  $c$  of the word  $w$  in sentence  $\mathbf{x}$  is not changed if sentence  $\mathbf{x}$  appears in any domain corpus, i.e.,  $P(c|\mathbf{x})$  does not depend on a domain. As a result,  $P_s(c|\mathbf{x}) = P_t(c|\mathbf{x})$ . Therefore, it seems that we do not need to estimate  $P_t(c|\mathbf{x})$  because we have  $P_s(c|\mathbf{x})$ . However, this is wrong because  $P_s(\mathbf{x}) \neq P_t(\mathbf{x})$ . The following assumption is referred to as the covariate shift:

$$P_s(\mathbf{x}) \neq P_t(\mathbf{x}), \quad P_s(c|\mathbf{x}) = P_t(c|\mathbf{x}).$$

In other words, the domain adaptation for WSD satisfies the assumption of the covariate shift. In this paper, we solve domain adaptation for WSD by learning under covariate shift.

Briefly, learning under covariate shift is a learning method through weighted training data. Thus, it has

<sup>1</sup>Domain adaptation is considered as a type of transfer learning (Kamishima, 2010) in part of machine learning.

two key points: (1) calculation of the weight of an instance and (2) weighted learning.

For the first point, the probability density ratio  $w(\mathbf{x}) = P_t(\mathbf{x})/P_s(\mathbf{x})$  is used theoretically as the weight of the instance  $\mathbf{x}$ . There are two techniques for calculating the probability density ratio. The first is modeling  $P_S(\mathbf{x})$  and  $P_T(\mathbf{x})$  and then taking the ratio between them. The second is modeling  $w(\mathbf{x})$  directly. Several studies have examined the former method (Jiang and Zhai, 2007)(Saiki et al., 2008). However, to the best of our knowledge, the latter approach has not been attempted in NLP research. In this paper, we adopt unconstrained least squares importance fitting (uLSIF) as the second calculation (Kanamori et al., 2009). Actually, there are many methods to calculate probability density ratio (Sugiyama and Kawanabe, 2011). In this paper, we use uLSIF because it shows good performance and quick calculation time. uLSIF models  $w(\mathbf{x})$  with the sum of  $N_t$  pieces of basis functions  $\psi_l(\mathbf{x})$ , where  $N_t$  is the number of target data.

$$w(\mathbf{x}) = \sum_{l=1}^{N_t} \alpha_l \psi_l(\mathbf{x}).$$

Generally, a Gaussian kernel is used as the basis function. However, in this case, the width  $\sigma$  of the Gaussian kernel becomes an additional parameter. Therefore, we suggest using a linear kernel to drop this parameter  $\sigma$ .

For the second point, the maximum entropy method (ME) is commonly employed in weighted learning. However, in domain adaptation for WSD, the number of instances is generally small. For this reason, we do not use a weighted ME but a weighted support vector machine (SVM).

Furthermore, three rough heaviness values are applied to the weighted SVM for comparison, i.e., a small weight 0.1, a normal weight 1.1, and a large weight 2.1, rather than a detailed weight for each case.

In the experiment, we use three domains, i.e., OC (Yahoo! Answer), PB (books) and PN (newspaper) in the Balanced Corpus of Contemporary Written Japanese (BCCWJ (Maekawa, 2007)) and 16 target words that appear frequently in these three domains. There are six types of domain adaptation - OC  $\rightarrow$  PB, PB  $\rightarrow$  PN, PN  $\rightarrow$  OC, OC  $\rightarrow$  PN,

PN  $\rightarrow$  PB, and PB  $\rightarrow$  OC giving a total of 96 (= 16  $\times$  6) domain adaptation tasks. Consequently, the effects of the proposed method are confirmed.

## 2 Related Work

Generally, methods for domain adaptation can be divided into instances-based method and features-based method (Pan and Yang, 2010). The instances-based method is a learning method that gives weight to an instance of training data. Learning under covariate shift is typical method for this type. The features-based method is a method that maps the source and target features spaces to a common features space to maintain the important characteristics in each domain by reducing the difference between domains. The paper (Blitzer et al., 2006) proposed the dimension reduction method called structural correspondence learning (SCL). The paper (Pan et al., 2008) evaluated the distance between the spaces mapped in the source domain and the spaces mapped in the target domain by maximum mean discrepancy (MMD). They proposed a conversion method to minimize the distance called MMD embedding (MMDE). Moreover, the paper (Pan et al., 2011) improved MMDE and proposed a novel method called transfer component analysis (TCA). Adding weight to features is a features-based method. The paper (Daumé III, Hal, 2007) offered a weighting system for features. In this study, vector  $\mathbf{x}_s$  of the training data in the source domain is mapped to an augmented input space  $(\mathbf{x}_s, \mathbf{x}_s, \mathbf{0})$ , and  $\mathbf{x}_t$  is mapped to an augmented input space  $(\mathbf{0}, \mathbf{x}_t, \mathbf{x}_t)$ . The classifier that learned from the augmented vectors solves the classification problem by the usual method. Daumé’s method assumes that an effect can be determined by overlapping the characteristics that are common to the source and target domains.

The domain adaptation problem is considered a data-sparse problem. Self-training and semi-supervised learning (Chapelle et al., 2006) and active learning (Settles, 2010) (Rai et al., 2010) are useful for domain adaptation.

At last, we introduce researches on domain adaptation for WSD. We assume that  $P_t(c|\mathbf{x}) = P_s(c|\mathbf{x})$ , but the assumption  $P_t(\mathbf{x}|c) = P_s(\mathbf{x}|c)$  is also possible. Under this assumption, we can solve domain adaptation for WSD by estimating  $P_t(c)$ . Ac-

tually, the papers (Chan and Ng, 2006) and (Chan and Ng, 2005) estimated  $P_t(c)$  by using EM algorithm to do it. The papers (Komiya and Okumura, 2012a), (Komiya and Okumura, 2011) and (Komiya and Okumura, 2012b) changed the learning method by the combination of source domain, target domain and target word. These studies are a kind of ensemble learning. In those learning methods, only the weight that is applied to data in source and target domain is different.

### 3 Domain Adaptation under Covariate Shift

In this section, we show that weighted learning can solve a domain adaptation problem under assumption of covariate shift.

We define the loss function as  $l(\mathbf{x}, c, d)$  where  $\mathbf{x}$ ,  $c$  and  $d$  denote an instance, the class of  $\mathbf{x}$  and a classifier respectively. Thus, expected loss function  $L_0$  in our task is expressed as the following:

$$L_0 = \sum_{\mathbf{x}, c} l(\mathbf{x}, c, d) P_T(\mathbf{x}, c).$$

Through the assumption of covariate shift, we obtain the following:

$$\frac{P_T(\mathbf{x}, c)}{P_S(\mathbf{x}, c)} = \frac{P_T(\mathbf{x})P_T(c|\mathbf{x})}{P_S(\mathbf{x})P_S(c|\mathbf{x})} = \frac{P_T(\mathbf{x})}{P_S(\mathbf{x})}.$$

Now,  $w(\mathbf{x}) = P_T(\mathbf{x})/P_S(\mathbf{x})$ . It establishes as follows:

$$L_0 = \sum_{\mathbf{x}, c} w(\mathbf{x})l(\mathbf{x}, c, d)P_S(\mathbf{x}, c).$$

$D = \{(\mathbf{x}_i, c_i)\}_{i=1}^N$  denotes the training data. Using empirical distribution as a substitute for  $P_S(\mathbf{x}, c)$ , the following holds

$$L_0 \approx \frac{1}{N} \sum_{i=1}^N w(\mathbf{x}_i)l(\mathbf{x}_i, c_i, d).$$

In terms of expected loss minimization, find  $d$  minimizing the following equation  $L_1$  to solve the problem of covariate shift.

$$L_1 = \sum_{i=1}^N w(\mathbf{x}_i)l(\mathbf{x}_i, c_i, d). \quad (1)$$

Consider a classification based on a posterior probability maximizing estimation.

$$d(\mathbf{x}) = \arg \max_c P_T(c|\mathbf{x}).$$

Additionally, adapt a logarithmic loss as a loss function. Eq. (1) turn into the following:

$$L_1 = - \sum_{i=1}^N w(\mathbf{x}_i) \log P_T(c_i|\mathbf{x}_i).$$

In case of adopting an approach using a model of  $P_T(c|\mathbf{x}, \boldsymbol{\lambda})$  in order to solve the classification problem, we find the parameter  $\boldsymbol{\lambda}$  maximizing the weighted log-likelihood  $L(\boldsymbol{\lambda})$  of the following weighted by the probability density ratio in covariate shift.

$$L(\boldsymbol{\lambda}) = \sum_{i=1}^N w(\mathbf{x}_i) \log P_T(c_i|\mathbf{x}_i, \boldsymbol{\lambda}). \quad (2)$$

For above problem, Maximum Entropy Method (ME) is commonly used as a model.

$$P_T(c|\mathbf{x}, \boldsymbol{\lambda}) = \frac{1}{Z(\mathbf{x}, \boldsymbol{\lambda})} \exp \left( \sum_{j=1}^M \lambda_j f_j(\mathbf{x}, c) \right), \quad (3)$$

where  $\mathbf{x} = (x_1, x_2, \dots, x_M)$ . The function  $f_j(\mathbf{x}, c)$  is a feature function. It returns  $x_j$  when the true class is defined  $c$ , and it returns 0 in other cases.  $Z(\mathbf{x}, \boldsymbol{\lambda})$  is a normalization term. Hence, we obtain

$$Z(\mathbf{x}, \boldsymbol{\lambda}) = \sum_{c \in C} \exp \left( \sum_{j=1}^M \lambda_j f_j(\mathbf{x}, c) \right), \quad (4)$$

where  $\boldsymbol{\lambda} = (\lambda_1, \lambda_2, \dots, \lambda_M)$  is a vector of weight parameters for features.

### 4 Weight through Probability Density Ratio

There are two kinds of approaches estimating the probability density ratio  $w(\mathbf{x}) = P_T(\mathbf{x})/P_S(\mathbf{x})$ . The first approach is estimating each  $P_S(\mathbf{x})$  and  $P_T(\mathbf{x})$ , and take their ratio. The second approach is modeling  $w(\mathbf{x})$ , directly.

In this paper, we use unconstrained least-squares importance fitting (uLSIF) proposed in (Kanamori et al., 2009) as the second approach.

#### 4.1 uLSIF

$\{\mathbf{x}_i^s\}_{i=1}^{N_s}$  and  $\{\mathbf{x}_i^t\}_{i=1}^{N_t}$  denote a source data and a target data, respectively. In uLSIF, the probability density ratio is modeled as the following:

$$w(\mathbf{x}) = \sum_{l=1}^{N_t} \alpha_l \psi_l(\mathbf{x}) = \boldsymbol{\alpha} \cdot \boldsymbol{\psi}(\mathbf{x}),$$

where  $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \dots, \alpha_{N_t})$ ,  $\boldsymbol{\psi}(\mathbf{x}) = (\psi_1(\mathbf{x}), \psi_2(\mathbf{x}), \dots, \psi_b(\mathbf{x}))$ . and  $\alpha_l > 0$ . Here,  $\psi_l(\mathbf{x})$  is a basis function which is mapping from the source data to the positive real number.

In uLSIF, actually, the parameter  $\boldsymbol{\alpha}$  is estimated after building the basis function  $\boldsymbol{\psi}(\mathbf{x})$ . However, for the convenience of description, we firstly explain the estimation of  $\boldsymbol{\alpha}$ .  $\hat{w}(\mathbf{x})$  denotes a model of  $w(\mathbf{x})$ . In order to estimate parameter  $\alpha_l$ , we find  $\hat{\alpha}$  minimizing a mean square error  $J_0(\boldsymbol{\alpha})$  between  $w(\mathbf{x})$  and  $\hat{w}(\mathbf{x})$ . By taking account of  $w(\mathbf{x}) = P_T(\mathbf{x})/P_S(\mathbf{x})$ ,  $J_0(\boldsymbol{\alpha})$  can be transformed as follows:

$$\begin{aligned} J_0(\boldsymbol{\alpha}) &= \frac{1}{2} \int (\hat{w}(\mathbf{x}) - w(\mathbf{x}))^2 P_S(\mathbf{x}) d\mathbf{x} \\ &= \frac{1}{2} \int \hat{w}(\mathbf{x})^2 P_S(\mathbf{x}) d\mathbf{x} \\ &\quad - \int \hat{w}(\mathbf{x}) w(\mathbf{x}) P_S(\mathbf{x}) d\mathbf{x} \\ &\quad + \frac{1}{2} \int w(\mathbf{x})^2 P_S(\mathbf{x}) d\mathbf{x} \\ &= \frac{1}{2} \int \hat{w}(\mathbf{x})^2 P_S(\mathbf{x}) d\mathbf{x} \\ &\quad - \int \hat{w}(\mathbf{x}) P_T(\mathbf{x}) d\mathbf{x} \\ &\quad + \frac{1}{2} \int w(\mathbf{x})^2 P_S(\mathbf{x}) d\mathbf{x}. \end{aligned}$$

Since the third term is constant, so it is independent on minimizing  $J_0(\boldsymbol{\alpha})$ . Therefore, minimizing  $J_0(\boldsymbol{\alpha})$  means minimizing the following  $J(\boldsymbol{\alpha})$ .

$$J(\boldsymbol{\alpha}) = \frac{1}{2} \int \hat{w}(\mathbf{x})^2 P_S(\mathbf{x}) d\mathbf{x} - \int \hat{w}(\mathbf{x}) P_T(\mathbf{x}) d\mathbf{x}.$$

By approximating  $P_S(\mathbf{x})$  and  $P_T(\mathbf{x})$  by empirical distributions,  $J(\boldsymbol{\alpha})$  is transformed as the following  $\hat{J}(\boldsymbol{\alpha})$ :

$$\hat{J}(\boldsymbol{\alpha}) = \frac{1}{2N_s} \sum_{i=1}^{N_s} \hat{w}(\mathbf{x}_i^s)^2 - \frac{1}{N_t} \sum_{j=1}^{N_t} \hat{w}(\mathbf{x}_j^t)$$

$$\begin{aligned} &= \frac{1}{2} \sum_{l,l'=1}^{N_t} \alpha_l \alpha_{l'} \left( \frac{1}{N_s} \sum_{i=1}^{N_s} \psi_l(\mathbf{x}_i^s) \psi_{l'}(\mathbf{x}_i^s) \right) \\ &\quad - \sum_{l=1}^{N_t} \alpha_l \left( \frac{1}{N_t} \sum_{j=1}^{N_t} \psi_l(\mathbf{x}_j^t) \right) \\ &= \frac{1}{2} \boldsymbol{\alpha}^T \hat{H} \boldsymbol{\alpha} - \hat{h}^T \boldsymbol{\alpha}, \end{aligned} \quad (5)$$

where  $\hat{H}$  denotes  $N_t \times N_t$  matrix, and  $\hat{H}_{l,l'}$  (the  $(l, l')$  element of  $\hat{H}$ ) is defined as follows:

$$\hat{H}_{l,l'} = \frac{1}{N_s} \sum_{i=1}^{N_s} \psi_l(\mathbf{x}_i^s) \psi_{l'}(\mathbf{x}_i^s)$$

Furthermore,  $\hat{h}$  denotes  $N_t$ -dimensional vector, and the element of the  $l$ -th dimension  $\hat{h}_l$  is defined as follows:

$$\hat{h}_l = \frac{1}{N_t} \sum_{j=1}^{N_t} \psi_l(\mathbf{x}_j^t).$$

As the result, we can obtain the  $\hat{\alpha}$  minimizing  $\hat{J}(\boldsymbol{\alpha})$  by solving the following problem:

$$\min_{\boldsymbol{\alpha}} \left[ \frac{1}{2} \boldsymbol{\alpha}^T \hat{H} \boldsymbol{\alpha} - \hat{h}^T \boldsymbol{\alpha} + \frac{\lambda}{2} \boldsymbol{\alpha}^T \boldsymbol{\alpha} \right].$$

Here, we must note that the parameter  $\lambda$  is added. The above minimization problem is unconstrained convex quadratic programming problem without a constrained condition, so that we obtain a global solution:

$$\tilde{\boldsymbol{\alpha}} = (\hat{H} + \lambda I_{N_t})^{-1} \hat{h}^T. \quad (6)$$

Lastly, conduct the following adjustment to satisfy the condition  $\boldsymbol{\alpha} > 0$ :

$$\begin{aligned} \hat{\boldsymbol{\alpha}} &= ((\max(0, \tilde{\alpha}_1), \max(0, \tilde{\alpha}_2), \dots, \max(0, \tilde{\alpha}_{N_t}))) \\ &= \max(0_{N_t}, \tilde{\boldsymbol{\alpha}}). \end{aligned} \quad (7)$$

In general, a Gaussian kernel is used as the basis function.

$$\psi_l(\mathbf{x}) = K(\mathbf{x}, \mathbf{x}_l^t) = \exp \left( -\frac{\|\mathbf{x} - \mathbf{x}_l^t\|^2}{\sigma^2} \right).$$

Under this situation, remaining parameters to be determined are the regularization term  $\lambda$  and a width of the Gaussian kernel  $\sigma$  to obtain the probability

density ratio. These parameters are found by a cross-validation of a grid search. First, split each source and target data into  $R$  pieces of subset with no intersection. Secondly, exclude the  $r$ -th subset from these subsets, and bind the rest. These data are regarded as a new source and target domain. Now, set certain values to  $\lambda$  and  $\sigma$ , and obtain  $\alpha$  with Eq. (6) and Eq. (7), and find  $\hat{J}(\alpha)^{(r)}$  with Eq. (5). Calculate  $R$  pieces of values of  $\hat{J}(\alpha)^{(r)}$  by varying the value  $r$  from 1 to  $R$ , and regard the average value of them as  $\hat{J}(\alpha)$  for  $\lambda$  and  $\sigma$ . Next, estimate  $\lambda$  and  $\sigma$  minimizing  $\hat{J}(\alpha)$  obtained with the above procedure by changing the values of  $\lambda$  and  $\sigma$ . These values are denoted by  $\hat{\lambda}$  and  $\hat{\sigma}$ , respectively.

#### 4.2 Use of Linear Kernel instead of Gaussian Kernel

In this paper, we use linear kernel as the basis function in uLSIF instead of the Gaussian kernel. By this use, we can drop the parameter  $\sigma$ .

Generally, a kernel function is to map to non-linear high-dimensional space. However, in our tasks, the number of features is larger than the number of instances, so that there is no need to map to the high-dimensional space. In this case, calculation to adjust the parameter is easier than using Gaussian kernel.

$$\psi_l(\mathbf{x}) = K(\mathbf{x}, \mathbf{x}_l^t) = \mathbf{x} \cdot \mathbf{x}_l^t.$$

#### 4.3 Weight of Particular Instances

In our task, that is domain adaptations of WSD, we must construct the model of the probability density for each target word. Additionally, the number of instances of the target word is too small compared to the number of the feature dimension in both source and target domain. Therefore, an estimated probability density ratio tends to be smaller than the true value, so that some approaches to close the estimated probability density ratio to 1 have been proposed. Sugiyama translated to the weight  $w$  to the weight  $w^p$  ( $0 < p < 1$ ) (Sugiyama, 2006), and Yamada proposed the relative probability density ratio (Yamada et al., 2011):

$$\frac{P_T(\mathbf{x})}{\alpha P_T(\mathbf{x}) + (1 - \alpha)P_S(\mathbf{x})}.$$

These methods have an effect to close the original probability density ratio to 1.

Here, we assign the rough weight to the instance according to the estimated probability density ratio. The rough weight has 3 kinds of value: 0.1, 1.1 and 2.1.

The set of the estimated probability density ratio is expressed as  $W = \{w_i\}_{i=1}^N$ . The  $w_i$  is normalized by the following:

$$w'_i = \frac{w_i - \mu}{\sigma},$$

where  $\mu$  and  $\sigma^2$  denote the mean and the variance of  $W$ , respectively.

Assuming that  $W$  follows a normal distribution,  $w_i$  is defined as 2.1 when  $w_i$  is greater than 0.84,  $w_i$  is defined as 0.1 when  $w_i$  is smaller than -0.84, and in the other cases,  $w_i$  is defined as 1.1.

The points, 0.84 and -0.84, are 20% top and lower quantile points of normal distribution, respectively.

### 5 SVM for weighted learning

Learning under covariate shift means the weighted learning using the probability density ratio. After assigning weight to each instance, we apply the weighted learning method. Conventionally we use ME and logistic regression as the the weighted learning method. However, a method based on a loss function is also available, as can be seen from the Eq. (1). In this paper, we use the method of SVM for imbalanced data (Tang et al., 2009). Through the training data  $\{(\mathbf{x}_i, y_i)\}_{i=1}^N$  ( $\mathbf{x}_i \in R^d, y_i \in \{1, -1\}$ ), SVM is constructed by estimating parameters  $\mathbf{w}$ ,  $b$ , and  $\zeta$  in the following:

$$\min_{\mathbf{w}, b, \zeta} \left\{ \frac{1}{2} \mathbf{w}^T \mathbf{w} + C \sum_{i=1}^N \zeta_i \right\}. \quad (8)$$

Now,

$$y_i(\mathbf{w}^T \phi(\mathbf{x}_i) + b) \geq 1 - \zeta_i, \quad \zeta_i \geq 0.$$

In the above formula, we can use the weighted SVM by using  $w(\mathbf{x}_i)C$  instead of  $C$  (Cortes and Vapnik, 1995).

### 6 Experiment

In this paper, we chose three domains, OC (Yahoo! Answer), PB (books), and PN (newspaper) in the BCCWJ, and 16 target words with enough frequency

Table 1: Target words

word	dictionary # of senses	OC freq. of word	OC # of senses	PB freq. of word	PB # of senses	PN freq. of word	PN # of senses
iu(言う)	3	666	2	1114	2	363	2
ireru(入れる)	3	73	2	56	3	32	2
kaku(書く)	2	99	2	62	2	27	2
kiku(聞く)	3	124	2	123	2	52	2
kodomo(子供)	2	77	2	93	2	29	2
jikan(時間)	4	53	2	74	2	59	2
jibun(自分)	2	128	2	308	2	71	2
deru(出る)	3	131	3	152	3	89	3
toru(取る)	8	61	7	81	7	43	7
baai(場合)	2	126	2	137	2	73	2
hairu(入る)	3	68	4	118	4	65	3
mae(前)	3	105	3	160	2	106	4
miru(見る)	6	262	5	273	6	87	3
motsu(持つ)	4	62	4	153	3	59	3
yaru(やる)	5	117	3	156	4	27	2
yuku(ゆく)	2	219	2	133	2	27	2
average	3.44	148.19	2.94	199.56	3.00	75.56	2.69

in all three domains. Now, we have six types of domain adaptation: OC  $\rightarrow$  PB, PB  $\rightarrow$  PN, PN  $\rightarrow$  OC, OC  $\rightarrow$  PN, PN  $\rightarrow$  PB, and PB  $\rightarrow$  OC. Therefore, totally 96 (= 6  $\times$  16) kinds of domain adaptation tasks is set.

Table 1 indicates information of the target word, the number of senses registered in the dictionary, and the number of senses and the frequency in each corpus<sup>2</sup>.

Here, we explain how to evaluate a domain adaptation method for our tasks. First, by using a domain adaptation method (named as Mtd-A), a classifier for a target word  $w_i$  in a domain adaptain  $S \rightarrow T$  is obtained. We can get the precision  $p_{w_i}^{(ST)}$  of this classifier under this setting. Thus, given domain adaptain  $S \rightarrow T$ , we can get the average precision  $p^{(ST)}$  for the 16 target words ( $w_1, w_2, \dots, w_{16}$ ). By this  $p^{(ST)}$ , we evaluate the method Mtd-A for the domain adaptain  $S \rightarrow T$ . We have 6 types of  $S \rightarrow T$ , so 6  $p^{(ST)}$  are obtained. We evaluate the method Mtd-A by taking average of 6  $p^{(ST)}$ .

The method Mtd-A is composed of two approaches, a method of calculating the probability

<sup>2</sup>The word “入る (hairu)” has three senses in the dictionary, but there are four senses in OC and PB. This is because our used sense tagged corpus accepts new senses.

density ratio and type of the weighted learning. In this paper, 10 kinds of method are examined. Base-M and Base-S, are approaches without uLSIF. The characters, “M” and “S”, mean ME and SVM, respectively. The other techniques are with uLSIF. The letters, “G” and “L” signify the Gaussian kernel and the linear kernel used in uLSIF, respectively. In addition, in Ours-\*.\*, convert weight into three types of weight (0.1, 1.1, and 2.1) depending on the probability density ratio calculated with uLSIF. Our proposed method is expressed as “Ours-L-S”.

Results of the experiments are shown in Table. 2. As the result, relationships, Base-M < Base-S, Mtd-G-M < Mtd-G-S, Mtd-L-M < Mtd-L-S, Ours-G-M < Ours-G-S, and Ours-L-M < Ours-L-S are satisfied. It is found that SVM is more effective than ME. Additionally, relations, Mtd-G-M < Mtd-L-M, Mtd-G-S = Mtd-L-S, and Ours-G-S < Mtd-L-S are established. The results of Ours-G-M and Ours-L-M are almost the same. Therefore, the linear kernel has better effectiveness than the Gaussian kernel. The proposed method Ours-L-S in this paper has the highest average accuracy rate. In each domain adaptation, it shows the highest accuracy rate, excluding PN  $\rightarrow$  PB.

Here, we must note that the difference between our proposed method (Ours-L-S) and the baseline (Base-S) is slight, and we could not get statistical sig-

Table 2: Experimental results (average precisions)

	OC $\rightarrow$ PB	PB $\rightarrow$ PN	PN $\rightarrow$ OC	OC $\rightarrow$ PN	PN $\rightarrow$ PB	PB $\rightarrow$ OC	Average
Base-M	0.7163	0.7700	0.6920	0.6778	<b>0.7474</b>	0.6991	0.7171
Base-S	0.7141	0.7676	0.6907	0.6880	0.7452	0.7011	0.7178
Mtd-G-M	0.7008	0.7289	0.6854	0.6840	0.7110	0.6760	0.6977
Mtd-G-S	0.7143	0.7692	0.6903	0.6900	0.7455	0.7034	0.7189
Mtd-L-M	0.7145	0.7339	0.6907	0.6887	0.7144	0.7008	0.7055
Mtd-L-S	0.7134	0.7699	0.6905	0.6898	0.7450	0.7045	0.7189
Ours-G-M	0.7145	0.7670	0.6907	0.6787	0.7446	0.7008	0.7160
Ours-G-S	0.7129	0.7707	0.6911	0.6884	0.7451	0.7021	0.7184
Ours-L-M	0.7145	0.7665	0.6907	0.6787	0.7445	0.7008	0.7159
Ours-L-S (Proposed Method)	<b>0.7197</b>	<b>0.7723</b>	<b>0.6971</b>	<b>0.6936</b>	0.7416	<b>0.7062</b>	<b>0.7218</b>

nificance. However, without taking account on the PN domain, our proposed method is statistical significant for the baseline. Further the use of weighted SVM is also significant for the use of weighted ME. From these points, our proposed method has its value.

## 7 Discussion

### 7.1 Effectiveness of Small and Large Weights

Our proposed method converts the weight estimated by uLSIF to 0.1, 1.1 or 2.1 according to the volume of the weight. In this section, we investigate which is effective small weight 0.1 or large weight 2.1. To do it, we modify our proposed method by following two cases: (case.1) In our method, we change the weight 2.1 to the normal weight 1.1, and other weights are not changed. (case 2) In our method, we change the weight 0.1 to the normal weight 1.1, and other weights are not changed.

We conducted experiments of the above two modification. The result is shown in Table 3. ‘‘Ours-L-S-small’’ and ‘‘Ours-L-S-large’’ in Table 3 denote (case 1) and (case 2), respectively.

This result shows that small weight 0.1 is more effective than large weight 2.1 in our proposed method, because the (case 2) is worse than baseline but the (case 1) is better than baseline. However, our proposed method is better than the (case 1). That is, the use of both weights is more effective than only small weight or only large weight.

### 7.2 Deletion of Misleading Data

In the previous section, we mentioned that small weight is effective, that is, it is effective to decrease the weight of unimportant training data in our task. The reason comes from that there are misleading data in the training data. Misleading data is a problem of domain adaptation. In domain adaptation,

Table 3: Importance of instances

	Average
Base-S	0.7178
Ours-L-S	0.7218
Ours-L-S-small (only small weight, case 1)	0.7183
Ours-L-S-large (only large weight, case 2)	0.7176

some data in training data decrease the precision of the classifier. These data is called misleading data (Jiang and Zhai, 2007).

In this section, we discuss the relation of our proposed method and misleading data. First, we confirm the presence of misleading data in training data. To do it, Yoshida (Yoshida and Shinnou, 2014) checked each training data is misleading data or not one by one.

Here, we introduce the above Yoshida’s method. In the domain adaptation from the source domain  $S$  to the target domain  $T$ , labeled data  $D$  in  $S$  of the target word  $w$  exists. First, we measure the precision  $p_0$  for  $T$  by the classifier learned through  $D$ . Second, we remove a data  $x$  from  $D$  and measure the precision  $p_1$  for  $T$  by the classifier learned through  $D - x$ . In the case of  $p_1 > p_0$ , the data  $x$  is regarded as the misleading data. We apply this procedure for all data in  $D$  to find the misleading data of the target word  $w$  in the domain adaptation from  $S$  to  $T$ . The result of the number of misleading data is shown in Table 4. The number in parentheses is the total number of the training data. Note that this method uses labels in  $T$ , so it cannot detect misleading data. This

Table 4: Number of the misleading data

word	OC → PB	PB → PN	PN → OC	OC → PN	PN → PB	PB → OC
iu(言う)	159 (666)	75 (1114)	82 (363)	158 (666)	35 (363)	127 (1114)
ireru(入れる)	6 (73)	15 (56)	3 (32)	28 (73)	1 (32)	19 (56)
kaku(書く)	21 (99)	2 (62)	12 (27)	39 (99)	15 (27)	0 (62)
kiku(聞く)	26 (124)	0 (123)	4 (52)	21 (124)	27 (52)	26 (123)
kodomo(子供)	5 (77)	1 (93)	12 (29)	0 (77)	13 (29)	12 (93)
jikan(時間)	1 (53)	0 (74)	0 (59)	8 (53)	5 (59)	0 (74)
jibun(自分)	13 (128)	0 (308)	0 (71)	25 (128)	1 (71)	0 (308)
deru(出る)	14 (131)	32 (152)	22 (89)	10 (131)	10 (89)	39 (152)
toru(取る)	6 (61)	18 (81)	12 (43)	5 (61)	22 (43)	10 (81)
baai(場合)	0 (126)	13 (137)	14 (73)	0 (126)	9 (73)	7 (137)
hairu(入る)	36 (68)	27 (118)	27 (65)	11 (68)	42 (65)	38 (118)
mae(前)	8 (105)	1 (160)	15 (106)	5 (105)	2 (106)	10 (160)
miru(見る)	10 (262)	12 (273)	8 (87)	3 (262)	28 (87)	3 (273)
motsu(持つ)	8 (62)	11 (153)	1 (59)	0 (62)	1 (59)	2 (153)
youtu(やる)	0 (117)	0 (156)	0 (27)	0 (117)	0 (27)	0 (156)
yuku(ゆく)	17 (219)	1 (133)	3 (27)	0 (219)	3 (27)	15 (133)

Table 5: Deletion of the misleading data

	OC → PB	PB → PN	PN → OC	OC → PN	PN → PB	PB → OC	Average
Base-S	0.7141	0.7676	0.6907	0.6880	0.7452	0.7011	0.7178
Ours-L-S	0.7197	0.7723	0.6971	0.6936	0.7416	0.7062	0.7218
Mislead	0.7459	0.7927	0.7450	0.7213	0.7869	0.7334	0.7542
Mislead2	0.7117	0.7627	0.6833	0.6920	0.7399	0.6984	0.7146

method is just to confirm the presence of misleading data.

The result of SVM using the training data without misleading data is shown in Table 5. “Mislead” in Table 5 denotes the average accuracy rate. This result is highest in our experiments. To remove of the misleading data is to assign the weight of the data to 0. Therefore, it is possible to improve the precision by just adjusting the weight.

Now, we conduct the experiment that the data with quite small probability density ratio is regarded as the misleading data whose weight is 0. The “Mislead2” in Table 5 shows the result. However, this approach is not effective. Probably, we cannot detect the misleading data using only the probability density ratio. The method to detect misleading data is our future work.

## 8 Conclusion

We have solved domain adaptation for WSD by learning under covariate shift. This learning has two

key points: (1) calculation of the weight of an instance and (2) weighted learning. For the first point, we used uLSIF and improved it by weighting only the particular instances and by using a linear rather than a Gaussian kernel in uLSIF. For the second point, we used a weighted SVM rather than the commonly used weighted ME.

Three corpora in BCCWJ and 16 target words (96 domain adaptation tasks) were used in our experiment. This experimental results show that the proposed method demonstrates the highest average precision. The proposed method is statistically significant for the baseline without considering the PN domain. In addition, the use of the weighted SVM is significant for the weighted ME.

In future, we will investigate why weighted learning does not work well for the  $PN \rightarrow PB$  domain adaptation.

## References

- John Blitzer, Ryan McDonald, and Fernando Pereira. 2006. Domain adaptation with structural correspondence learning. In *EMNLP-2006*, pages 120–128.
- Yee Seng Chan and Hwee Tou Ng. 2005. Word sense disambiguation with distribution estimation. *IJCAI-05*.
- Yee Seng Chan and Hwee Tou Ng. 2006. Estimating class priors in domain adaptation for word sense disambiguation. In *COLING-ACL-2006*, pages 89–96.
- Olivier Chapelle, Bernhard Schölkopf, Alexander Zien, et al. 2006. *Semi-supervised learning*, volume 2. MIT press Cambridge.
- Corinna Cortes and Vladimir Vapnik. 1995. Support-vector networks. *Machine learning*, 20(3):273–297.
- Daumé III, Hal. 2007. Frustratingly Easy Domain Adaptation. In *ACL-2007*, pages 256–263.
- Jing Jiang and Chengxiang Zhai. 2007. Instance weighting for domain adaptation in nlp. In *ACL-2007*, pages 264–271.
- Toshihiro Kamishima. 2010. Transfer learning (in japanese). *The Japanese Society for Artificial Intelligence*, 25(4):572–580.
- Takafumi Kanamori, Shohei Hido, and Masashi Sugiyama. 2009. A least-squares approach to direct importance estimation. *The Journal of Machine Learning Research*, 10:1391–1445.
- Kanako Komiya and Manabu Okumura. 2011. Automatic Determination of a Domain Adaptation Method for Word Sense Disambiguation using Decision Tree Learning. In *IJCNLP-2011*, pages 1107–1115.
- Kanako Komiya and Manabu Okumura. 2012a. Automatic Domain Adaptation for Word Sense Disambiguation Based on Comparison of Multiple Classifiers. In *PACLIC-2012*, pages 75–85.
- Kanako Komiya and Manabu Okumura. 2012b. Automatic selection of domain adaptation method for wsd using decision tree learning (in japanese). *Journal of NLP*, 19(3):143–166.
- Kikuo Maekawa. 2007. Design of a Balanced Corpus of Contemporary Written Japanese. In *Symposium on Large-Scale Knowledge Resources (LKR2007)*, pages 55–58.
- Sinno Jialin Pan and Qiang Yang. 2010. A survey on transfer learning. *Knowledge and Data Engineering, IEEE Transactions on*, 22(10):1345–1359.
- Sinno Jialin Pan, James T Kwok, and Qiang Yang. 2008. Transfer learning via dimensionality reduction. In *AAAI*, volume 8, pages 677–682.
- Sinno Jialin Pan, Ivor W Tsang, James T Kwok, and Qiang Yang. 2011. Domain adaptation via transfer component analysis. *Neural Networks, IEEE Transactions on*, 22(2):199–210.
- Piyush Rai, Avishek Saha, Hal Daumé III, and Suresh Venkatasubramanian. 2010. Domain adaptation meets active learning. In *NAACL HLT 2010 Workshop on Active Learning for Natural Language Processing*, pages 27–32.
- Yosuke Saiki, Hiroya Takamura, and Manabu Okumura. 2008. Domain adaptation in sentiment classification by instance weighting (in japanese). *IPSJ SIG Technical Report. SIG-NL Report*, 2008(33):61–67.
- Burr Settles. 2010. Active learning literature survey. *University of Wisconsin, Madison*.
- Masashi Sugiyama and Motoaki Kawanabe. 2011. *Machine Learning in Non-Stationary Environments: Introduction to Covariate Shift Adaptation*. MIT Press.
- Masashi Sugiyama. 2006. Supervised learning under covariant shift (in japanese). *Japanese Neural Network Society*, 13(3):111–118.
- Yuchun Tang, Yan-Qing Zhang, Nitesh V Chawla, and Sven Krasser. 2009. SVMs modeling for highly imbalanced classification. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, 39(1):281–288.
- Makoto Yamada, Taiji Suzuki, Takafumi Kanamori, Hirotaka Hachiya, and Masashi Sugiyama. 2011. Relative density-ratio estimation for robust distribution comparison. *Neural Computation*, 25(5):1370–1370.
- Hiromu Yoshida and Hiroyuki Shinnou. 2014. Detection of misleading data by outlier detection methods (in japanese). In *The 5th Japanese Corpus Linguistics Workshop*, pages 49–56.

# Unsupervised Domain Adaptation for Word Sense Disambiguation using Stacked Denoising Autoencoder

**Kazuhei Kouno, Hiroyuki Shinnou, Minoru Sasaki, Kanako Komiya**

Ibaraki University, Department of Computer and Information Sciences

4-12-1 Nakanarusawa, Hitachi, Ibaraki JAPAN 316-8511

15nm707g@vc.ibaraki.ac.jp

{hiroyuki.shinnou.0828, minoru.sasaki.01, kanako.komiya.nlp}  
@vc.ibaraki.ac.jp

## Abstract

In this paper, we propose an unsupervised domain adaptation for Word Sense Disambiguation (WSD) using Stacked Denoising Autoencoder (SdA). SdA is an unsupervised learning method of obtaining the abstract feature set of input data using Neural Network. The abstract feature set absorbs the difference of domains, and thus SdA can solve a problem of domain adaptation. However, SdA does not always cope with any problems of domain adaptation. Especially, difficulty of domain adaptation for WSD depends on the combination of a source domain, a target domain and a target word. As a result, any method of domain adaptation for WSD has adverse effect for a part of the problem. Therefore, we defined the similarity between two domains, and judge whether we use SdA or not through this similarity. This approach avoids an adverse effect of SdA. In the experiments, we have used three domains from the Balanced Corpus of Contemporary Written Japanese and 16 target words. In comparison with baseline, our method has got higher average accuracies for all combinations of two domains. Furthermore, we have obtained better results against conventional domain adaptation methods.

## 1 Introduction

In this paper, we propose an unsupervised method of domain adaptation for Word Sense Disambiguation (WSD) using Stacked Denoising Autoencoder (SdA).

WSD is the task of identifying the sense of a target word in a sentence. In general, supervised learning,

such as Support Vector Machine (SVM), can be used for this task because of the fact that this approach is highly accurate. However, if the training and test data come from different domains, the accuracy of this approach is lowered. This problem is called a *domain adaptation* (Søgaard, 2013). It is considered that this problem occurs due to the difference between the distributions of features in training and test data.

SdA is an unsupervised learning method of obtaining the abstract feature of the input data (basic feature) using Neural Network (Vincent et al., 2010). Recently it has been shown that a higher accuracy in voice and character recognition has been obtained using SdA (Le et al., 2012). We have applied this method to a domain adaptation for WSD and have shown that the abstract feature obtained through SdA can avoid the problem of domain adaptation.

It is well-known from previous works that the most efficient methods for domain adaptation for WSD depend on the combination of training data (from the source domain) and test data (from the target domain) (Komiya and Okumura, 2011) (Komiya and Okumura, 2012). Furthermore, in an unsupervised domain adaptation method, even if the accuracy is improved in the combination of the source and target domains, the accuracy rate hardly improve. As a result, the accuracy rate on average of the method decreases, or remains the same. In other words, there are accuracy limitations with each method. In our method, we choose whether or not to apply SdA based on the similarity of features. Our method cannot be applied in the case for pair of do-

mains as they are not suitable for SdA.

In our experiment, we have used three domains: Yahoo! Answers (OC), Books (PB), and newspaper (PN) from the Balanced Corpus of Contemporary Written Japanese (Maekawa, 2007), along with 16 selected ambiguous words. Domain adaptation has the following six transitions: (1) PB  $\rightarrow$  OC, (2) OC  $\rightarrow$  PB, (3) OC  $\rightarrow$  PN, (4) PN  $\rightarrow$  OC, (5) PB  $\rightarrow$  PN and (6) PN  $\rightarrow$  PB. First, in every domain adaptation, we have compared the accuracy of the basic feature and abstract feature by SdA using SVM. As a result, SdA have been effective in half of the case of domain adaptations. Furthermore, we have explored situations when to apply SdA or not. Consequently, the SdA with similarity of features is effective in all domain adaptations.

## 2 Domain Adaptation for WSD

Frequently, the word has multiple senses. Word Sense Disambiguation (WSD) is the task of identifying a sense of the such word in a sentence.

In general, supervised learning like SVM can be used for this task, because this approach shows a high accuracy. However, in these methods, training and test data must come from same domain. In the case of WSD, these are often obtained from different domains. For example, to learn the classifier using sentences from books as training data, and then classify the word in the sentence from newspaper. In this case, it can't well identify the test data from newspaper (target domain) by the classifier which is learned by books (source domain). To solve this problem, tuning the classifier that is learned by training data from source domain to match the test data from target domain is necessary. It is called *domain adaptation* (Søgaard, 2013).

It is considered that this problem occurs from the difference between distributions of features in training and test data. Therefore, we attempt to absorb it by SdA.

## 3 Related Work

Inductive learning is used not only WSD but also many natural language processing tasks, and domain adaptation problem will occur. There are two types of methods for this problem. One is a supervised domain adaptation using labeled data in the target do-

main and the other is an unsupervised domain adaptation that does not use it. Typically, in the domain adaptation tasks, supervised and semi-supervised learning show the high accurate (Chapelle et al., 2006). However, supervised learning are inappropriate in WSD because they use labeled data of target domain, even though the data of target domain is new data. Although semi-supervised learning requires many data of target domain, the data for each target word is not so many in WSD. Therefore, unsupervised learning is appropriate in domain adaptation for WSD. SdA for use in this study is an unsupervised learning method, and our method can be classified into unsupervised domain adaptation.

As research on unsupervised domain adaptation, there are structural correspondence learning (SCL) (Blitzer et al., 2006) and learning under covariate shift (Sugiyama and Kawanabe, 2011). In SCL, measure the mutual information from label and feature value; features the value is large are elected to Pivot feature. Features to co-occur with Pivot feature are used for classification. This is based on the idea that Pivot features are different depending on the domain, in contrast, feature to co-occur with Pivot feature are effective in classification. Learning under covariate shift is regarded as weighted learning, where sentence  $x$  is weighted with the probability density ratio  $w(x) = P_T(x)/P_S(x)$ . There are many methods to calculate probability density ratio. In this paper, we adopt unconstrained least squares importance fitting (uLSIF) (Kanamori et al., 2009) because it shows good performance and quick calculation time.

These approaches depend on the combination of source and target domain; there is also case that accuracy is going to negative. As a result, accuracy rate has been decreased, or dose not develop on average.

## 4 Stacked Denoising Autoencoder

SdA is an unsupervised learning method of obtaining the abstract feature of input data (basic feature) by using Neural Network. SdA is composed of multiple Denoising Autoencoder (dA). As mentioned above, domain adaptation for WSD has a problem that the accuracy is lowered from the difference between distributions of features in training and test

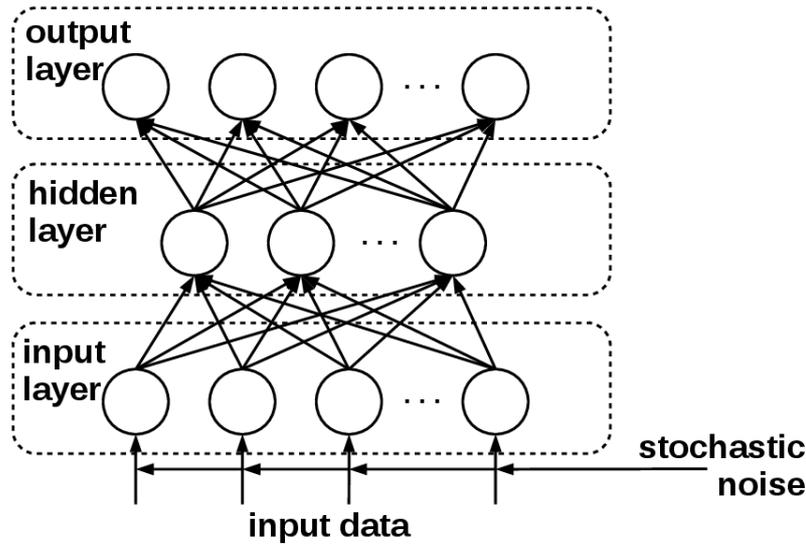


Figure 1: Denoising Autoencoder

data. The abstract feature obtained through SdA can avoid the problem of domain adaptation.

#### 4.1 Denoising Autoencoder

The dA has input layer, hidden layer and output layer, as shown in Figure 1. At first, append stochastic noise to the input data and transmit to the input layer. Then, the data on the input layer is encoded and transmitted to the hidden layer. Similarly, data on the hidden layer is decoded and transmitted to the output layer. In this model, to learn the encoder and decoder, such as error of input data (without noise) and output layer becomes smaller. In other words, dA learns the model, such as to eliminate the noise that was added at first.

Number of nodes in the input and output layer are equal to the dimensions of input data. Typically, number of nodes in the hidden layer is set to be smaller than other layers. If the input data  $x = \{x_1, x_2, \dots, x_N\}$  and input layer with noise  $\tilde{x}$ , mapping from the input layer  $\tilde{x}$  to the hidden layer  $y$ , and from the hidden layer  $y$  to the output layer  $z$  are represented by the following formula  $y, z$ .

$$y = \sigma(W\tilde{x} + b)$$

$$z = \sigma(W^T y + b')$$

where  $b, b', W$  and  $W^T$  indicate bias, another bias, the weight matrix and the transposed matrix of  $W$  respectively. The  $\sigma(\cdot)$  indicates sigmoid function:

$$\sigma(x) = \frac{1}{1 + e^{-x}}$$

Finding the  $W$  (or  $W^T$ ),  $b$  and  $b'$ , such as mean squared error is minimized using a Stochastic Gradient Descent(SGD). Hidden layer  $y$  obtained in this process is the abstract feature of the input data  $x$ , because it can be restored the input data by decoder; nevertheless number of nodes is less.

#### 4.2 Stacked Denoising Autoencoder

SdA is a model of stacked multiple dA, as shown in Figure 2. At first, to learn using dA that the input is the input data (call  $dA_1$ ). Then, to learn using dA that the input is hidden layer of  $dA_1$ (call  $dA_2$ ). In  $dA_3$ , the input is hidden layer of  $dA_2$ ; SdA stacks learning by repeating this process. In this way, the abstract feature is gradually obtained from the input data. Note that output layers for each dA are used only to calculate the mean squared error; mainly, hidden layers are used on SdA.

In this paper, connecting input data and the abstract feature, to absorb the difference in distributions of features between training and test data.

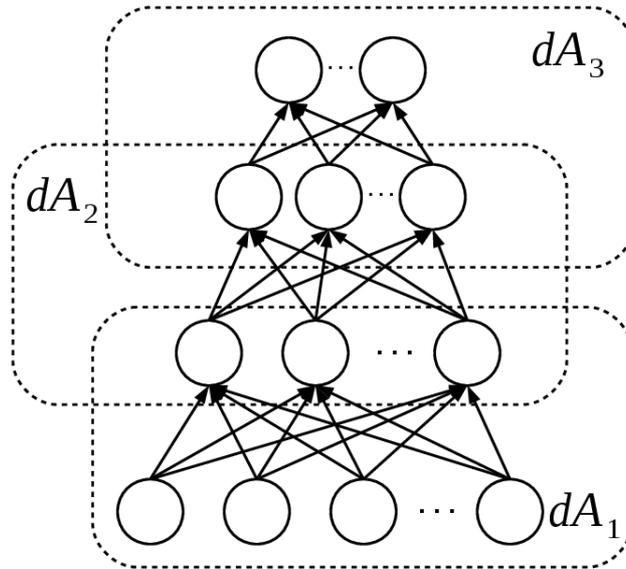


Figure 2: Stacked Denoising Autoencoder

Specifically, to extract  $n_{abst}$ -dimensional abstract feature  $x_{abst}$  from  $n$ -dimensional input data  $x$  by SdA, and then,  $x$  and  $x_{abst}$  are separately normalized. We use the data that  $x$  and  $x_{abst}$  are connected to classification by SVM.

### 5 Similarity of feature

In chapter 3, we introduced previous studies using SCL or uLSIF, as unsupervised domain adaptation. These approaches depend on the combination of source and target domain and there are also cases that accuracy is going to negative. As a result, accuracy rate has been decreased, or dose not develop on average. In other words, the best method for domain adaptation for WSD depends on the combination of source and target domain. Therefore, we choose whether to apply SdA based on the combination of source domain, target domain and a target word.

Configuring small number of nodes in the hidden layer than the other layers; SdA reduces dimension of data. SdA is expected that to project distributions of features from source and target domain. If both training and test data have little commonality, SdA requires a lot of data to learn a model. Typically, it

is not possible to learn a better model, since training and test data are less in WSD. Therefore, to calculate similarity of feature, and then apply the SdA if this value is large.

While cosine similarity and mutual information are typical as a way to measure the similarity, we use simple approach that calculate the ratio of the number of common dimensions to all dimensions. Specifically, to determine occurrence vector of dimensions  $\vec{S}$  and  $\vec{T}$  for the training data  $X_S$  and test data  $X_T$ , and then to calculate the similarity  $P_f$  by following equations:

$$P_f = \frac{\vec{T} \cdot \vec{S}}{n}$$

dimension of training data  $X_S$  and test data  $X_T$  are  $dim\vec{S}$  and  $dim\vec{T}$ , respectively. Where  $dim\vec{S} = dim\vec{T}$  is satisfied; there are represented as  $n$ . If  $P_f$  is greater than the threshold  $T$ , it is regarded that training and test data have some commonality, and then apply SdA.

Table 1: Target words

word	dictionary # of senses	OC freq. of word	OC # of senses	PB freq. of word	PB # of senses	PN freq. of word	PN # of senses
iu(言う)	3	666	2	1114	2	363	2
ireru(入れる)	3	73	2	56	3	32	2
kaku(書く)	2	99	2	62	2	27	2
kiku(聞く)	3	124	2	123	2	52	2
kodomo(子供)	2	77	2	93	2	29	2
jikan(時間)	4	53	2	74	2	59	2
jibun(自分)	2	128	2	308	2	71	2
deru(出る)	3	131	3	152	3	89	3
toru(取る)	8	61	7	81	7	43	7
baai(場合)	2	126	2	137	2	73	2
hairu(入る)	3	68	4	118	4	65	3
mae(前)	3	105	3	160	2	106	4
miru(見る)	6	262	5	273	6	87	3
motsu(持つ)	4	62	4	153	3	59	3
yaru(やる)	5	117	3	156	4	27	2
yuku(ゆく)	2	219	2	133	2	27	2
average	3.44	148.19	2.94	199.56	3.00	75.56	2.69

## 6 Experiment

### 6.1 Data and Methods

In the experiment, we compare the effect by following methods:

- baseline: classify the basic feature by SVM
- uLSIF
- SCL
- SdA
- proposed method : SdA using similarity

We use the data from the Balanced Corpus of Contemporary Written Japanese (BCCWJ (Maekawa, 2007)) that has word sense tags by a Japanese WSD SemEval-2 task (Okumura et al., 2010). Among them, we use three domains as different domains: Yahoo! Answers (OC), Books (PB) and Newspaper (PN). Table 1 indicates information of the target word, the number of senses registered in the dictionary, and the number of senses and the frequency in each corpus<sup>1</sup>. All methods learn the

<sup>1</sup>The word “入る (hairu)” has three senses in the dictionary,

classifier using the training data from source domain; and then, classify the test data from target domain by the classifier (as represented by  $source \rightarrow target$ ). There are six domain adaptation patterns: (1) PB  $\rightarrow$  OC, (2) OC  $\rightarrow$  PB, (3) OC  $\rightarrow$  PN, (4) PN  $\rightarrow$  OC, (5) PB  $\rightarrow$  PN and (6) PN  $\rightarrow$  PB. There are six domain adaptations and sixteen target words; the experiments are made 96 ways. We evaluated each methods by following. First, to calculate the accuracy rate for each combination of source domain, target domain and target word. Then, to calculate the average for each domain adaptation. Similarly, to calculate average of 96 pairs; they are accuracy of each methods. In the proposed method, threshold  $T$  of similarity is equal to 0.2; if  $P_f > 0.2$ , then we choose to apply SdA.

In this experiment, we use 8 kinds of features for a sentence, that is an instance. They are shown in Table 2, where  $w$  and  $w_i$  represent target word and the  $i$ -th word from the word  $w$  respectively.

but there are four senses in OC and PB. This is because our used sense tagged corpus accepts new senses.

Table 3: Average accuracy rate (%)

Domain Adaptation	baseline	uLSIF	SCL	SdA	our method
OC → PB	71.33	71.34	71.34	71.09	<b>71.43</b>
PB → OC	70.10	70.45	70.18	71.01	<b>70.93</b>
OC → PN	68.81	68.98	<b>69.24</b>	68.18	68.81
PN → OC	69.09	69.05	68.94	67.49	<b>69.24</b>
PB → PN	76.76	76.99	76.65	<b>77.33</b>	77.02
PN → PB	74.55	74.50	73.47	<b>75.37</b>	74.59
average	71.77	71.89	71.64	71.74	<b>72.00</b>

Table 2: feature of sentence

feature	content
(e0)	written of $w$
(e1)	parse of $w$
(e2)	written of $w_{-1}$
(e3)	parse of $w_{-1}$
(e4)	written of $w_1$
(e5)	parse of $w_1$
(e6)	written of independent word between $w_{-3}$ and $w_3$
(e7)	Number from classification vocabulary table of $e6$ (4 and 5-digit)

### 6.2 Parameters of SdA

We use Pylearn2<sup>2</sup> for learning the model of SdA. The number of repetitions of dA is twice. In  $dA_1$  (Input is input data.), when the dimension of the input data is  $N$ , the number of nodes of hidden layer is  $2/3 \times N$ . In  $dA_2$  (Input is hidden layer of  $dA_1$ ), the number of nodes of hidden layer is equal to input layer's, that is following equation:

$$\begin{aligned}
 DimOfInput &= InputLayerOf dA_1 \\
 &= \frac{2}{3} \times HiddenLayerOf dA_1 \\
 &= \frac{2}{3} \times InputLayerOf dA_2 \\
 &= \frac{2}{3} \times HiddenLayerOf dA_2
 \end{aligned}$$

where as stated above, the number of nodes in output layer are equal to input layer's. On this calculation,

<sup>2</sup><http://deeplearning.net/software/pylearn2/>

round the result to an integer.

The hidden layer of  $dA_2$  are connected to the basic feature, and then classified using SVM. Where basic and abstract feature are respectively normalized before connection. We use libsvm<sup>3</sup> as classification by SVM; kernel function is linear kernel that is often used in natural language processing tasks. Similarly, baseline also uses libsvm with linear kernel.

### 6.3 Results

Table 3 shows the result of our experiments.

In uLSIF, accuracy are improved in four domain adaptations, and on average, it's above the baseline. However, it was opposite effect for two domain adaptations. SCL and SdA also has good and bad results. Consequently, three methods were not much different. Meanwhile, proposed method showed high accuracy in five domain adaptations; there was no bad result in all domain adaptations. As a result, our proposed method shows best accuracy among all methods.

## 7 Discussions

In each domain adaptation, method that showed the best accuracy among the four methods baseline, uLSIF, SCL and SdA are shown in Table 4. The best method is different depending on the domain adaptation. Moreover, baseline showed the best result in PN → OC. This results suggest effectiveness of selecting the method by any way.

<sup>3</sup><http://www.csie.ntu.edu.tw/~cjlin/libsvm/>

Table 4: Best method for each domain adaptation

Domain Adaptation	Method
OC $\rightarrow$ PB	uLSIF, SCL
PB $\rightarrow$ OC	SdA
OC $\rightarrow$ PN	SCL
PN $\rightarrow$ OC	baseline
PB $\rightarrow$ PN	SdA
PN $\rightarrow$ PB	SdA

In this study, we bring in similarity of features, and choose whether to apply the SdA depending on the combination of training data, test data and target word. As a results, accuracy has improved in five domain adaptations, compared with baseline. In the other one domain adaptation, it shows improvement in the third decimal place. Our method showed a better result than the other four methods on average.

However, our method has a problem to be solved. Proposed method chooses either baseline or SdA for each combination of source domain, target domain and target word. If the pair is improved by SdA that does not use similarity  $P_f$ , improvement has decreased in our method compared to SdA. If our method rise to the same level as SdA in these pairs, it can be expected to more improve on average. The following two methods will be considered to achieve it.

1. Decreasing the threshold  $T$ .
2. If  $P_f$  is less than  $T$ , to modify parameter of SdA.

In approach 1, selectivity of SdA is increased by decreasing  $T$ . As a result, we expect that proposed method is close to the accuracy of SdA. However, if  $T$  is extremely low, the proposed method will show the same results as SdA. In the previous experiments, the  $T$  is equal to 0.2. There are experiments that the  $T$  is lowered to 0.18. The results are shown in Table 5.

Out of three domain adaptation ( PB  $\rightarrow$  OC, PB  $\rightarrow$  PN and PN  $\rightarrow$  PB) that is impaired with the our method compared to SdA, accuracy has improved in two domain adaptation (PB  $\rightarrow$  PN and PN  $\rightarrow$  PB) by lowering  $T$ . Moreover, it shows better results on PN  $\rightarrow$  PB than the SdA. However, it

is worse on PB  $\rightarrow$  OC than the case of  $T = 0.2$ . About these results, we consider the influence of decision to apply the SdA for each pair of word and domains. Besides, in the case of  $T = 0.18$ , two domain adaptation have a poor accuracy as compared to baseline. Nevertheless, the method which is 0.18 shows the best results on average. For this reason, it is necessary to determine the appropriate threshold  $T$ .

In approach 2, if the similarity of feature  $P_f$  is fewer than the threshold  $T$ , we modify parameter of SdA. Consequently, SdA will get the feature close to the basic feature; the result is close to SdA. The parameter to be adjusted include the number of nodes in hidden layer for each dA, and the number of repetitions of dA. If the number of nodes in the hidden layer is increased, there is no difference between the dimensions of basic feature and abstract feature; SdA gets abstract features similar to basic feature. If, however, the number of nodes in hidden layer is large, learning requires a long time, because the bonds between each nodes are increased. Furthermore, learning data is not so much in WSD, there is not enough learning. An approach of increasing the number of repetitions of dA has also same problems, because the first dA have to set the large number of nodes. For this reason, if we have enough data and times, this approach is effective.

## 8 Conclusions

In this paper, we have proposed an unsupervised method of domain adaptation for WSD using SdA. Specifically, the basic features are converted to abstract features by SdA, and then, these are classified by SVM.

In the domain adaptation methods for WSD, the most powerful method is different from each other depending on the pair of source and target domains; there are also accuracy limitations within each method. In this paper, we have introduced a similarity of the features and the option of choosing whether to apply SdA or not.

In our experiments, we chose three domains and 16 selected ambiguous words. While uLSIF, SCL and SdA have shown poor accuracy in some case of domain adaptation, our method has been a better accuracy in all situations of domain adaptation and

Table 5: Average accuracy rate on additional experiment (%)

Domain Adaptation	baseline	SdA	our method ( $T = 0.2$ )	our method ( $T = 0.18$ )
OC → PB	71.33	71.09	<b>71.43</b>	71.31
PB → OC	70.10	71.01	<b>70.93</b>	70.66
OC → PN	68.81	68.18	68.81	<b>68.91</b>
PN → OC	69.09	67.49	<b>69.24</b>	68.85
PB → PN	76.76	<b>77.33</b>	77.02	77.12
PN → PB	74.55	75.37	74.59	<b>76.02</b>
average	71.77	71.74	72.00	<b>72.14</b>

had a better result as compared with other methods. In our future work, we plan to examine pair of domains where our method has not performed well as compared with SdA that dose not use similarity.

### References

- John Blitzer, Ryan McDonald, and Fernando Pereira. 2006. Domain Adaptation with Structural Correspondence Learning. In *EMNLP-2006*, pages 120–128.
- Olivier Chapelle, Bernhard Schölkopf, Alexander Zien, et al. 2006. *Semi-supervised learning*, volume 2. MIT press Cambridge.
- Takafumi Kanamori, Shohei Hido, and Masashi Sugiyama. 2009. A Least-Squares Approach to Direct Importance Estimation. *The Journal of Machine Learning Research*, 10:1391–1445.
- Kanako Komiya and Manabu Okumura. 2011. Automatic Determination of a Domain Adaptation Method for Word Sense Disambiguation using Decision Tree Learning. In *IJCNLP-2011*, pages 1107–1115.
- Kanako Komiya and Manabu Okumura. 2012. Automatic Domain Adaptation for Word Sense Disambiguation Based on Comparison of Multiple Classifiers. In *PACLIC-2012*, pages 75–85.
- Quoc Le, Marc’Aurelio Ranzato, Rajat Monga, Matthieu Devin, Kai Chen, Greg Corrado, Jeff Dean, and Andrew Ng. 2012. Building high-level features using large scale unsupervised learning. In *ICML-2012*.
- Kikuo Maekawa. 2007. Design of a Balanced Corpus of Contemporary Written Japanese. In *Symposium on Large-Scale Knowledge Resources (LKR2007)*, pages 55–58.
- Manabu Okumura, Kiyooki Shirai, Kanako Komiya, and Hikaru Yokono. 2010. SemEval-2010 Task: Japanese WSD. In *The 5th International Workshop on Semantic Evaluation*, pages 69–74.
- Anders Søgaard. 2013. *Semi-Supervised Learning and Domain Adaptation in Natural Language Processing*. Morgan & Claypool.
- Masashi Sugiyama and Motoaki Kawanabe. 2011. *Machine Learning in Non-Stationary Environments: Introduction to Covariate Shift Adaptation*. MIT Press.
- Pascal Vincent, Hugo Larochelle, Isabelle Lajoie, Yoshua Bengio, and Pierre-Antoine Manzagol. 2010. Stacked Denoising Autoencoders: Learning Useful Representations in a Deep Network with a Local Denoising Criterion. *The Journal of Machine Learning Research*, 11:3371–3408.

# Construction of Semantic Collocation Bank Based on Semantic Dependency Parsing

Liu Shijun<sup>1</sup>, Shao Yanqiu<sup>1\*</sup>, Zheng Lijuan<sup>1</sup>, Ding Yu<sup>2</sup>

<sup>1</sup>Information Science School, Beijing Language and Culture University, Beijing, China

<sup>2</sup>Computer Science and Technology School, Harbin Institute of Technology, Harbin, China

## Abstract

This paper extracts collocation basing on semantic dependency parsing, and then constructs a collocation bank with two levels according to frequency: the instance-level and semantic level. Compared with conventional extracting ways, the collocation extracted in this paper have closer relationship and higher quality both on the lexical structure and semantic structure.

## 1 Introduction

Collocation has always been an important issue in language research, especially in Chinese language researches. Chinese is an isolated language, which lacks morphological changes. Establishing a relatively complete dictionary of Chinese collocation will be a great contribution to Chinese study and research.

Collocation plays a significant supporting role in many fields of NLP, such as information retrieval, machine translation, information extraction, and so on. Ding and Bai proposed a method of query expansion based on local co-occurrence<sup>[1]</sup>; Lin put relationship of collocation into language model for query expansion, which got over the deficiency of insufficient relationships caused by lacking context in tradition query<sup>[2]</sup>. In the basic

research field of NLP, such as syntax, semantics, etc., collocation also plays an important role. Based on the comparison of different patterns in adjective collocation between the Chinese English learners and native speakers, Zhang analyzed the typical characteristics of different learners when using adjective collocations<sup>[3]</sup>; Xing emphasized on the importance of collocation in the second language learning<sup>[4]</sup>.

The early research of automatic collocation extraction was made by Choueka, Klein and Neuwitz, they defined collocation as adjacent words, and used co-occurrence frequency to extract collocation<sup>[6]</sup>; Church and Hanks improved the automatic extraction technology and put forward mutual information as the index of collocation evaluating<sup>[7]</sup>. By proposing a formula for calculating strength between collocation, introducing dispersion formula, as well as integrating with the automatic speech tagging technology, the Xtract system of Smadja improved the extraction accuracy rate of collocation extraction up to 80%<sup>[8]</sup>; Lin extracted collocation based on shallow syntactic parsing<sup>[9]</sup>; Shouxun YANG applied the method of decision tree to extract collocation by integrating frequency, likelihood ratio, point mutual information, variance and other statistical indicators<sup>[10]</sup>.

In China, there were a number of outstanding dictionaries had been published,

---

\*Correspondent Author

for example, *Modern Chinese Notional Words Collocation Dictionary* composed by Lin and *Modern Chinese Collocation Dictionary* by Mei. Recently, based on the foreign research results, Sun started a research on automatic extraction of collocation and proposed three statistical indicators namely strength, dispersion and spike<sup>[11]</sup>; Sun used the rule-based method to identify the verb object structure<sup>[12]</sup>; Quproposed the framework-based method to extract collocation<sup>[13]</sup>; Che applied frequency, distance, and variance, using improved t-test method to get the value of “collocation intensity coefficient”, which was used to measure relationship of collocation<sup>[14]</sup>.

This paper adopted method based on semantic dependency parsing (SDP) to extract collocation, combining with the semantic information from “*HowNet*” to make a semantic classification of collocation, which is a new perspective of collocation study.

## 2 Redefinition of Collocation on the Basis of Semantics

From the perspective of linguistics, the research done by Benson has been most influential in the field of collocation. This is the definition of collocation given in his famous book *BBI Combinatory Dictionary of English*: collocation is the combination of words with arbitrary and frequent co-occurrence. At present, most of the research on collocation is based on the above definition, introducing different statistics to express different features of collocation. For example: “frequent co-occurrence” can be calculated by “word frequency”, and “arbitrary” can be calculated by “mutual information”. But extracting by these statistical methods will lose important language information. Choueka extracted the adjacent words as collocation, missing the non-adjacent words such as “make...decision”; Church adopted MI as the feature to extract collocations, then words

which is closely related to each other but has no grammatical relations would interfere the results, such as “doctor-nurse”. Linguistic symbolises the combination of sound and semantic, referring to the psychological reality, which reflects the objective reality. Therefore, any two of the words with semantic relationship can express certain objective reality. Here are two types of special collocation need to be explained.

One is the so-called unusual collocation. Such as “土豪(tyrant)+金(gold)” (Chinese word, referring to gold iPhone 5S), originally this combination was not a collocation, but in recent years, with the popularity of Apple's mobile phone, this combination has become a common collocation. This is the evolution of language, which involves the principle of “established by usage”. The principle stipulates if the language phenomenon is widely accepted by language users, we should keep it as a common usage. Therefore, this paper choose the frequency to represent the principle and defined “土豪+金”, such a kind of frequent words, as collocation.

The other is free combination. Unlike constraint combination, the free combination is not combined in a relatively specific way; they can also be combined with other words, according to Benson. This kind of combination had been abandoned by Lin, when he was composing *Modern Chinese Collocation Dictionary* as they go against the principle of “less but better”<sup>[5]</sup>. For instance, the constraint combination such as “现代(modern)+词语(words), 古代(ancient)+词语(words)” had all been collected in this dictionary, but the free combination such as “好(good)+词语, 坏(bad)+词语” had not. This paper regarded them as a part of language, tried to find out their common semantic features, and generalized these combinations into the form of “word + semantic category”.

The collocation this paper defined has the

following characteristics: 1.there must be semantic dependency relations between words. 2. Reaching the threshold of frequency of co-occurrence. In this paper, there are two kinds of collocationforms: “word + word” and “word + Semantic Category”.

### 3 Corpus and Semantic Dictionary

Our research of collocation is based on SDP and semantic dictionary. Before the introduction of the bank, we should make a brief introduction to the SDP corpus we built and the semantic dictionary “HowNet” applied in this paper.

#### 3.1 Corpus of Semantic Dependency Graph(SDG)

Chinese is a parataxis languagewith flexible word order and diverse function of word class. In real language context Chinese

word often depends on several words simultaneously<sup>[12]</sup>, which means in the same sentence one word can be semantically related to several other words. It also may exist in the non-projection phenomenon of crossed arcs. These phenomenon cannot be explained by the traditional dependency trees<sup>[15]</sup>. In order to express these phenomenon and also take the advantages of dependency expression, this paper break through the limitation of the dependency tree and express the semantics of the sentence by using the dependency graph, namely we connected two words to a dependency arc as long as there is a semantic relationship between them, which means the situation that a word with multiple parent nodes and crossed arcs will be reasonable. For example, “她 (she) 眼睛 (eyes) 哭 (cry) 肿 (swollen) 了 (already)”, its dependency graph is shown in Fig. 1.

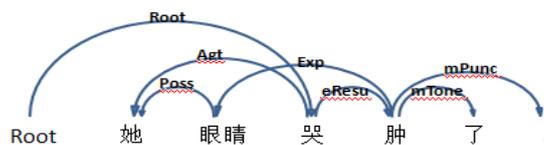


Fig. 1. An example of dependency graph

As shown in Fig. 1, the node “她 (she)” has semantic relations with both “哭 (cry)” and “眼睛 (eyes)”, which means that there are two heads for “她 (she)”: “哭 (cry)” and “眼睛 (eyes)”, separately indicates Agent of “哭 (cry)” and Possessor of “眼睛 (eyes)” Meanwhile, arcs (哭 (cry)->她 (she), Agt) and (肿 (swollen)->眼睛 (eyes), Exp)cross. Such a multi father node and the crossed arcs express the true

meaning of sentence.In addition, the meaning cannot be comprehensively expressed by dependency tree. “哭 (cry)” is the core word of whole sentence, and the result of dependency tree parsing is as follows: (哭->她, Agt), (她->眼睛, Bleg), (哭->肿, eResu), (肿->了, mTone) ,so as to lose the semantic relationship between "眼睛 (eyes)" and "肿 (swollen)". As in Fig. 2:



Fig. 2. An example of dependency tree

A set of semantic system has been constructed,

and this paper will make a brief introduction.

On the basis of this system, we built a semantic dependency graph corpus, which contains 30,000 sentences. We have completed correcting 10,038 sentences. These data are from different areas, including the news corpus (10,068), Chinese textbooks (10,038), Sina Weibo corpus (5,000) and corpus for machine

### 3.2 Corpus for Collocation Extraction

In order to reflect the truth of language more accurately, we chose a 4G news corpus to extract collocation. After carrying out the word segmentation and POS tagging to the corpus, we conducted the automatic SDP to the sentences. The results of the analysis are

translation (4,900). This semantic dependency graph database aims at solving some Chinese phenomenon perfectly by introducing the non-projective phenomenon, as well as improving the automatic semantic dependency tagging.

represented in the form of CoNLL data format<sup>[16]</sup>, which is shown in Table 1.

The meaning of the semantic tags in Table 1 are: Agt-agent, Poss-possessor, Exp-experiencer, Root-root (core word), eResu-result, mTone-tone mark.

Table 1. Table form of dependency graph

Word index	word	Part of speech	Head index	Head word	Semantic Role
1	她(she)	PN	2	眼睛(eyes)	Poss
1	她(she)	PN	3	哭(cry)	Agt
2	眼睛(eyes)	NN	4	肿(swollen)	Exp
3	哭(cry)	VV	Root	Root(root)	Root
4	肿(swollen)	VA	3	哭(cry)	eResu
5	了(already)	AS	4	肿(swollen)	mTone

### 3.3 HowNet

HowNet, built by Dong, is a knowledge base of common sense, aiming at revealing the relationship between the concepts and between the attributes of concept.

The semantic knowledge dictionary is the basic file system of HowNet, and the concept and description of the word in this dictionary form a record. Each record consists of several items. Each item has two parts separated by "=". The left part of the "=" represents the domain name of the data, the right is the value. Set the word "eye" as an example, the data recorded in HowNet is shown in Table 2. Each line in Table 2 represents a record item, the meanings of items are as follows: word index(No.), word(W\_C), part of speech and

pronunciation(G\_C), English explanation(W\_E), part of speech in English(G\_E), concept(DEF). The first position of the concept of DEF is the main characteristic, specified in HowNet, which is the most distinguishing characteristic from other words and usually expressed by the sememe. We extracted this main characteristic as the foundation for classifying semantic category. As shown in Table 2, we defined the semantic category of "眼睛(eyes)" as "part (部件)".

In this paper, we mainly investigated the collocation of "VV + NN" structure. With various relationships of entity in HowNet, we made a further classification to the nouns

(“NN”) in the instance collocation bank, generalizing the collocation into the form of

“VV + semantic category”, so as to establish the semantic collocation bank.

#### 4 Construction of Collocation Bank

##### 4.1 Framework of System

Table 2. An representation example of HowNet

NO.=131783
W_C=眼睛
G_C=N [yan3 jing1]
W_E=eye
G_E=N
DEF={part 部件:PartPosition={eye 眼},whole={AnimalHuman 动物}}

This paper takes the “VV+NN” as an example to build a collocation bank. There are two levels of bank; one is the instance collocation bank with collocation of “word + word”, which was extracted based on SDP, the other is semantic collocation bank with collocation of “word + semantic category”, which is the result from generalization of the instance bank based on HowNet. The system framework is shown in Figure 3. The upper section of the framework is the training part of the SDP model. We used this model to do SDP to the original corpus and got the semantic dependency graph. Then we extracted the ordered pairs with semantic dependency relations as the candidate set of collocation; on the basis of the candidate set, we extract the collocation of “VV + NN”, and introduce HowNet to construct the semantic collocation bank. Finally, the results of these two types of collocation database can be fed back to train the SDP model, to improve the effectiveness of the semantic dependency parsing system.

##### 4.2 Construction of Instance Collocation Bank

###### Bank

##### 4.2.1 Principles for Extracting the Semantic tuples

The structures of “NN + VV” and “VV +

NN”are different in traditional syntactic analysis. The former is a subject-predicate structure, while the latter is a verb-object structure. In semantic analysis, the parent node of the noun “NN”is verb“VV”in subject predicate structure and predicate object structure, and the direction of the dependency arcs is from verb to nouns, the dependency pairs are all “VV + NN” structure. For instance, “政府 (government) 打击 (beat) 盗版 (piracy)” and “盗版 (piracy) 被 (been) 政府 (government) 打击 (beat)”, in these two sentences, the extracted dependency pairs (打击 (beat), 政府 (government), Agt) 和 ( 打击 (beat), 盗版 (piracy), Pat) , are both in “VV + NN” structure. This paper mainly researches on the predicate-object structure, which can better reflect the collocation relationship between verbs and nouns compared with the subject-predicate.

Besides, Semantic level and syntactic level are not one to one correspondence, such as “吃 (eat) 食堂 (canteen)” and “在 (at) 食堂 (canteen) 吃 (eat)”, the former is the predicate object relationship, and the latter is the place adverbial. But after semantic dependency parsing, the arcs between them are both from verb “吃 (eat)” to noun “食堂 (canteen)”, shown as the tuple (吃 (eat), 食堂 (canteen), Loc), “Loc” refers to “location”.

According to the above analysis, we get rid of the ordered pairs which contains semantic subject roles, such as “Agt(agent), Poss(possessor), Aft(affection), Exp(experience), to obtain the candidate instances of “VV+NN” collocations in the predicate object structure.

#### 4.2.2 Steps for Extracting Collocation Instance

Step 1: According to the result of POS tagging and SDP, draw out the sets of tuple  $(W_{VV}, Role, W_{NN})$ . “ $W_{VV}$ ” and “ $W_{NN}$ ” respectively represents the words of verb and noun, “Role” represents the semantic

relationship between these two dependency words.

Step 2: Get rid of the tuple of which the “Role” remarked with subject role such as “Agt, Poss, Aft, Exp”, to get tuple in the predicate object structure as candidate collocation. In the meantime, as collocation is not only to meet the requirements of the semantics, but also to achieve a certain frequency, we set a threshold value for the frequency of co-occurrence, the pair with co-occurrence frequency  $> 50$  can be selected into our bank. Table 3 lists part of the collocations ranked in the top 20.

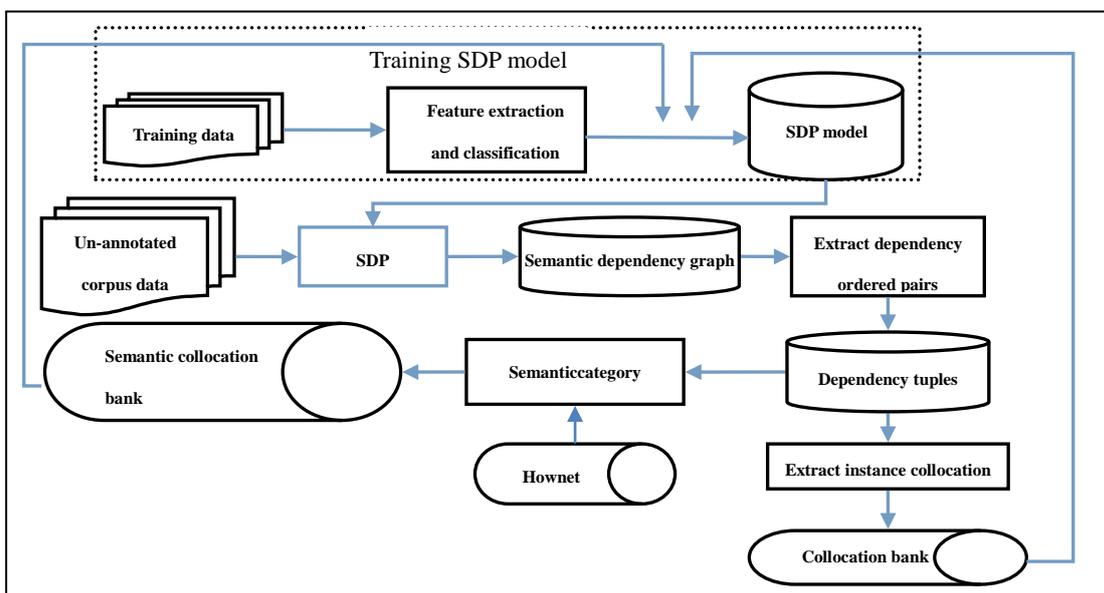


Fig. 3.system framework of collocation extraction

Table 3: part of the collocation ranked in the top 20

Verb	Noun	Semantic role	Frequency of co-occurrence
采取(take)	措施(measure)	Cont	4669
解决(solve)	问题(problem)	Pat	4473
出席(attend)	会议(meeting)	Cont	3469
举行(hold)	会谈(interview)	Cont	2921
充满(be filled with)	信心(confidence)	Cont	2910
发挥(play)	作用(role)	Cont	2251
交换(exchange)	意见(opinion)	Cont	2075
拉开(pull)	帷幕(curtain)	Cont	1886
处于(be)	状态(status)	Loc	1662
赶到(arrive)	现场(spot)	Dir	1537

Step 3: the same combination can be labeled with different semantic role by automatic SDP, we adopted probability to distinguish same pairs with different semantic labels. As shown in Table 4, the frequency of “争执(dispute)”marked as “Cont” in “发生(happen) + 争执(dispute)” is 181, marked as “Prod” is 21. We calculated the probability on the basis of frequency, and formed the collocation as the

structure of "W<sub>VV</sub>+ W<sub>NN</sub>+ semantic role + probability". After the semantic role's classification, the part of the "vocabulary + vocabulary" instance of collocation is shown in table 4.

In accordance with the above steps, this paper has a total of 67912 “VV+NN” structure as the collocation examples.

Table 4: instance bank of “word+word”

Semantic role	Verb	Noun	Probability
Prod	发生(happen)	争执(dispute)	0.10
Pat	限制(restrict)	自由(freedom)	0.26
	调查(investigate)	此案(case)	0.34
Datv	介绍(introduce)	总统(president)	1.00
	调查(investigate)	此案(case)	0.06
Cont	发生(happen)	争执(dispute)	0.90
	调集(assemble)	军队(army)	1.00
	举办(hold)	讲座(lecture)	1.00
	调查(investigate)	此案(case)	0.60
	限制	自由	0.74

### 4.3 Construction of Semantic Collocation

#### Bank

To make up the deficiency of data sparse of instance bank, we generalized “VV + NN” collocation in instance bank to construct “Word + semantic category” bank. This kind of semantic bank represents semantic relations between verb and semantic category. For example, “吃(eat)+ 食物(edible)”, “食物(edible)” refers to semantic category, this semantic collocation labeled with “Pat” can cover all collocation consisting of all edible nouns and the verb “吃(eat)”. We need to apply a semantic knowledge dictionary to determine whether a noun belongs to a certain semantic category. This paper introduced “HowNet”. The constructive algorithm of the semantic collocation bank is shown in Fig.3.

By algorithm 1, we generalized the semantic collocation from the candidate instance, and add the probability to each

semantic collocation in accordance with the method used in the extraction of instance collocation bank. The result is shown in Table 5. We totally extracted 1446 “word + semantic category” collocation.

### 5 Conclusions

On the basis of SDP, we extracted all the “VV + NN” ordered pairs from a large scale of news corpus, filtered collocation according to the set threshold and constructed the instance collocation bank. According to the semantic information provided by HowNet, we generalized the instance bank to conduct the semantic collocation bank. The relationship between the instance collocation bank and semantic collocation bank is complementary. The instance collocation bank cannot cover all the collocation in a language, so the semantic collocation is needed to be generalized, to enhance its robustness. For the extraction of collocation was based on SDP, the collocations are closely related to the syntactic structure and

semantic structure. Therefore, the quality of the collocations is better than those extracted by the traditional way.

In the actual process, we find that not all nouns can be classified, for example, “吃(eat) 定心丸(assurance)”, this kind of collocation lacks similarity, main feature of “定心丸(assurance)” in HowNet is “Text”. It is obvious that the nouns in “Text” category almost cannot

be matched with the verb “吃(eat)”. We put them into the instance collocation bank.

Two kinds of collocation bank have been automatically established in this paper. The quality of the bank has not been evaluated. Besides, these two banks can be used to improve the accuracy of SDP, which will be our next work.

```

algorithm 1: Constructing semantic bank
    input 1: set of tuple with semantic relation, each of which represented as  $(W_{VV}, W_{NN}, Role, Freq)$ , Role refers to the semantic labels of dependency pair  $(W_{VV}, W_{NN})$ , Freq refers to frequency of dependency pair with semantic label “Role”.
    Input 2: Hownet (as shown in Table 2) .
    Output: semantic collocation bank. Every item is in represented by the form of  $(W_{VV}, SemC, Role, Prob)$ , SemC refers to semantic category of nouns, Role refers to semantic role.
    Process:
    For each tuple in tuple set, Do{
    If  $W_{NN}$  in the tuple is in the Hownet:
        Add SemC represented by  $W_{NN}$  to the Four-dimensional tuple  $(W_{VV}, W_{NN}, Role, Freq)$ , extending the tuple into five-dimension  $(W_{VV}, W_{NN}, Role, Freq, SemC)$ 
    If  $Freq \geq 4$ 
        Replace the  $(W_{VV}, W_{NN}, Role, Freq, SemC)$  with  $(W_{VV}, SemC, Role, Prob)$  }
    
```

Fig. 3.algorithm 1

Table 5: semantic collocation bank

Verbs	SemC	Role	Prob	Verbs	SemC	Role	Prob
预防(prevent)	disease	Cont	1.000	去(go)	place	Dir	1.000
返回(return)	place	Dir	1.000	拘留(detain)	human	Pat	1.000
抵达(reach)	place	Cont	0.667	发展(develop)	Ability	Cont	1.000
带来(bring)	mishap	Cont	1.000	帮助(help)	human	Datv	1.000
看(look)	text	Cont	1.000	举办(hold)	place	Loc	1.000
	information	Cont	1.000	迎接(greet)	human	Datv	0.821
	shows	Cont	1.000	学习(study)	language	Cont	1.000
吃(eat)	vegetable	Pat	1.000	逮捕(arrest)	human	Pat	0.595
	edible	Pat	1.000	发出(emit)	sound	Cont	1.000
	fruit	Pat	1.000	公布(publish)	plans	Cont	1.000

**Acknowledgement**

We appreciatively acknowledge the support of the National Natural Science Foundation of China (NSFC) via Grant 61170144, Major Program of China’s National Linguistics work

Committee during the twelfth five-year plan (ZDI125-41), important special fund of Beijing Language and Culture University (13ZDY03) and young and middle aged academic cadre support plan of Beijing Language and Culture University (501321303).

## References

- [1] Ding Guodong, Baishuo, Wang Bin. Local Co-occurrence Based Query Expansion for Information Retrieval. *Journal of Chinese Information*, 2006, 20(3): 84-91.
- [2] Lin Jianfang. *Research on Collocation Extraction and Its Application in Information Retrieval*. Harbin: Harbin Institute of Technology, 2010.
- [3] Sun Haiyan. A Corpus-Based Study of Semantic Characteristic of Adjective Collocations in CLEC. *Modern Foreign Language*, 2004 (4).
- [4] Xing Hongbin. Collocation Knowledge and Second Language Lexical Acquisition. *Applied Linguistics*, 2013(4).
- [5] Lin Xingguang. *Research on Collocation*. *Chinese Teaching & Studies*, 1994(4).
- [6] Choueka, YandKlein, TandNeuwitz, E. Automatic Retrieval of Frequent Idiomatic and Collocational Expressions in A Large Corpus. *Journal of Literary and Linguistic Computing*, 1983,4.
- [7] K.church, P.Hanks. Word Association Norms.Mutual information and Lexicography. *Computational Linguistics*, 1990, 16 (1) : 22-29.
- [8] Smadja,F. Retrieving Collocation from Text : Xtract. *Computational Linguistic*, 1993, 19(1) : 143-177.
- [9] Lin D. Extracting Collocations from Text Corpora[C]. In *First Workshop on Computational Terminology*, Montreal, Canada, 1998: 8-12.
- [10] Yang S. Machine Learning for Collocation Identification. In *2003 IEEE International Conference on Natural Language Processing and Knowledge Engineering (NLP-KE 03)*. Beijing, 2003: 315-320.
- [11] Sun Maosong. Preliminary study on quantitative analysis of Chinese Collocation. *Studies of The Chinese Language*, 1997(1).
- [12] Sun Honglin. Generalizing Grammar Rules from Annotated Corpus: analysis of “V + N”. *The Forth China National Conference on Computational Linguistics*, 1997. Beijing: TsinghuaUniversity Press.
- [13] Qu Weiguang, Chen Xiaohe, Ji Genlin. A Frame-based Approach to Chinese Collocation Automatic Extracting. *Computer Engineering*, 2004.12.
- [14] CheWanxiang, Liu Ting, Qin Bing. A Method to Fetch Collocations Orienting Dependency Grammar. *The sixth China National Conference on Computational Linguistics*, 2001. Beijing: TsinghuaUniversity Press.
- [15] Zheng Lijuan, Shao Yanqiu, Yang Erhong. Analysis of the Non-projective Phenomenon in Chinese Semantic Dependency Graph. *Journal of Chinese Information Processing*, 2014.11.
- [16] Ding Yu, Shao Yanqiu, CheWanxiang, Liu Ting. *Dependency Graph based Chinese Semantic Parsing*. Harbin: Harbin Institute of Technology, 2014.

# Dynamic Semantics for Intensification and Epistemic Necessity: The Case of *Yídìng* and *Shìbì* in Mandarin Chinese

Wu, Jiun-Shiung

Chung Cheng University,  
168, University Road,  
Minshiung, Chiayi County,  
Taiwan, 621

Lngwuj@s@ccu.edu.tw

## Abstract

Functioning as adverbials, *yídìng* and *shìbì* in Mandarin Chinese can either express intensification or (strong) epistemic necessity. In addition, context influences their semantics. Hence, dynamic semantics are proposed for them. An information state  $\sigma$  is a pair  $\langle \mathcal{A}, s \rangle$ , where  $s$  is a proposition and  $\mathcal{A}$  is an affirmative ordering. *Yídìng*( $\phi$ ) performs update on an information state:  $\mathcal{A}$  is updated with  $\phi$  and  $s$  is specified to be a subset of or equal of  $\phi$ , as long as  $\phi$  is true in one of the absolutely affirmative worlds. Otherwise, uttering *yídìng*( $\phi$ ) leads to an absurd state. This is how a strong epistemic necessity reading is derived. To yield an intensification reading, *yídìng*( $\phi$ ) performs a test on the information state. *Yídìng*( $\phi$ ) gives back the original information state as long as  $\phi$  is true in all of the absolutely affirmative worlds. Otherwise, an absurd state is produced. As for *shìbì*, its semantics is identical to that of *yídìng*, except for that the  $s$  in an information state  $\sigma$  for *shìbì* is underspecified and needs resolving before a proposition gets an appropriate interpretation. The information needed to resolve the underspecified  $s$  for *shìbì* must be inferred from the context.

## 1 Introduction

In Mandarin Chinese (henceforth, Chinese), intensification and modal necessity can be expressed by the same lexical item. Adverbial *yídìng* is one of

such lexical items.<sup>1</sup> Please refer to the following examples.

- (1) A: Zhāngsān xǐhuān Xiǎoměi ma?  
Zhangsan like Xiaomei Q<sup>2</sup>  
'Does Zhangsan like Xiaomei?'
- B: Zhāngsān yídìng xǐhuān Xiǎoměi  
Zhangsan YÍDÌNG like Xiaomei  
Tā hěn zhùyì Xiǎoměi-de  
He very pay.attention.to Xiaomei-ASSO  
yìjùyídòng. Zhè shì hěn hélǐde  
move this be very reasonable  
tuīcè.  
conjecture  
'It must be the case that Zhangsan likes Xiaomei. He pays much attention to every move of Xiaomei. This is a reasonable conjecture.'
- B': Zhāngsān yídìng xǐhuān Xiǎoměi.  
Zhangsan YÍDÌNG like Xiaomei  
Zhè shì zhòngsuǒzhōngzhīde shìshí.  
This be widely-known fact  
'Zhangsan definitely likes Xiaomei. This is a widely-known fact.'

<sup>1</sup> Please note that *yídìng* can function either as a nominal modifier or a propositional modifier. The former is referred to as adjectival *yídìng* and the latter adverbial *yídìng*. This paper discusses adverbial *yídìng* only because the semantics of adjectival *yídìng* is simple and not as rich as adverbial *yídìng*.

<sup>2</sup> The abbreviations used in this paper include: ASSO for an associative marker, DEON for a deontic modal expression, DYN for a dynamic modal expression, EPI for an epistemic modal expression, Prc for a sentence-final particle, Prg for a progressive marker, Q for an interrogative particle.

(1) contains two conversations: one between A and B, and the other between A and B'. In the two conversations, A asks whether Zhangsan likes Xiaomei. Although the same sentence *Zhāngsān yídìng xīhuān Xiǎoměi* 'Zhangsan YÍDÌNG like Xiaomei' is uttered as a response to A's question, *yídìng* has different semantic functions. In the utterance of B, *yídìng* expresses epistemic necessity because B says that the proposition *Zhāngsān yídìng xīhuān Xiǎoměi* 'Zhangsan YÍDÌNG like Xiaomei' is a reasonable conjecture. *Yídìng* of this usage is translated as *it must be the case that...*

Moreover, when expressing epistemic necessity, *yídìng* expresses 'strong' epistemic necessity. The following examples demonstrate the difference between epistemic necessity and 'strong' epistemic necessity.

- (2) a. Ruóguǒ zài kǎo bù jǐgé, nǐ  
 If again take.exam not pass you  
 māma yídìng hěn shēngqì.  
 Mom YÍDÌNG very angry  
 'If you fail the exam again, it must be the case that you mom will be very angry.'
- b. Ruóguǒ zài kǎo bù jǐgé, nǐ  
 If again take.exam not pass you  
 māma huì hěn shēngqì.  
 Mom will very angry  
 'If you fail the exam again, you mom will be very angry.'

The difference between (2a) and (2b) lies in that (2a) contains *yídìng*, while (2b) uses *huì*. *Huì* has several meanings and one of them is inference, e.g. Chang (2000), Liu (1997), etc. In (2b), *huì* is used express an inference about a future situation based the antecedent led by *rúguǒ* 'if'. Although *yídìng* in (2a) has a similar function, (2a) and (2b) have a subtle semantic difference: (2a) shows a stronger certainty of the speaker's regarding the truth of the proposition *your Mom will be angry*, compared to (2b). Hence, when used to indicate an inference, *yídìng* is said to express 'strong' epistemic necessity.

On the other hand, *yídìng* in the utterance of B' has a different semantic function. In this utterance, *yídìng* is used to intensify the speaker's affirmativeness toward the proposition *your Mom will be angry*, instead of expressing the proposition as an inference. The intensification function of *yídìng* in this example is made explicit because of B' claims

that the proposition (= *Zhāngsān yídìng xīhuān Xiǎoměi* 'Zhangsan YÍDÌNG like Xiaomei') is a widely-known fact. This usage of *yídìng* is translated as *definitely* in English and is referred to as an intensification reading.

*Shìbì* has a semantic function similar to *yídìng* and they are interchangeable in some examples, but not in others. See below.

- (3) a. Yīnwèi zhùzi tài xì, yòng zhè zhǒng  
 Because pillar too thin, use this kind  
 wūdǐng yídìng/shìbì yǒu kěnéng  
 roof YÍDÌNG/SHÌBÌ have possibility  
 tāxiàlái.  
 collapse  
 'Because the pillars are too thin, if this type of roof is used, it is definitely possible that the roof will collapse.'
- b. Rúgǒu nǐ chuān hòu yīfú, nǐde  
 if you wear thick clothes your  
 shāng yídìng/shìbì jiào qīng.  
 wound YÍDÌNG/SHÌBÌ relatively minor  
 'If you wear thick clothes, it must be the case that your wound is relative minor.'
- (4) a. Zhè-ge shíhòu, Xiǎomíng yídìng/\*shìbì  
 This-CL time Xiaoming YÍDÌNG/\*SHÌBÌ  
 zài jiā.  
 at home  
 'At this moment, it must be the case that Xiaoming is at home.'
- b. Hūn hòu, rúguǒ zhù Yìnní, wǒ  
 married after if live Indonesia I  
 \*yídìng/shìbì cídiào gōngzuò.  
 \*YÍDÌNG/SHÌBÌ resign job  
 'After getting married, if we live in Indonesia, I definitely have to quit my job.'

In (3a, b), *yídìng* and *shìbì* are interchangeable and these two sentences are pretty much synonymous. However, in (4a, b), they are not interchangeable. In (4a), only *yídìng* is allowed, whereas in (4b) only *shìbì* is permissible.

In this paper, I would like to address the following questions. First, is it possible to provide a unified semantics for *yídìng* and *shìbì*? Second, how can the unified semantics account for the semantic similarity and difference between *yídìng* and *shìbì* as demonstrated in (3) and (4)? Finally, how can the unified semantics take care of contextual influence on the semantics of *yídìng* and *shìbì* illustrated by the utterances of B and of B' in (1)?

This paper is organized as follows. In Section Two, I critically review literature on *yídìng* and *shìbì*. In Section Three, I present more data and provide dynamic semantics for *yídìng* and *shìbì*. Section Four concludes this paper.

## 2 Review of Previous Studies

The literature on *yídìng* and/or *shìbì* include Chen (2011), Ding (2008a, b), C. Li (2005), S. Li (2009), Wang (2007), Xu (1995), Zhou (2014), etc. Xu (1995) is on the English translations of *yídìng* and two other adverbs and is not reviewed here. I critically review the other seven studies.

I start with the literature on *yídìng* and conduct the review in chronological order. Li (2005) distinguishes two variants of *yídìng*, labeled as *yídìng*<sub>1</sub> and *yídìng*<sub>2</sub>. He suggests that the former expresses strong volition, either the subject's or the speaker's strong volition (for another person) to do something, while the latter denotes stipulation or judgment. He further claims that *yídìng*<sub>1</sub> often goes with *yào*, which expresses a deontic reading here, or with *děi*, which also has a deontic reading, and that *yídìng*<sub>2</sub> often goes with *shì* 'be' or *hui*, which denotes epistemic necessity.

A major problem with Li (2005) is that he does not take the intensification reading into consideration, such as the utterance of B' in (1). Another problem is that the semantic contribution of *yídìng* is blurred when it goes with another modal expression. For example, he suggests that *yídìng děi* 'YÍDÌNG DEON' expresses a deontic reading. Then, a reasonable question to ask is what semantic contribution *yídìng* has here. The same problem occurs to *yídìng huì* 'YÍDÌNG EPI'.

Ding (2008a, b) also discusses the semantics of *yídìng*. These two studies distinguish *yídìng*<sub>1</sub> from *yídìng*<sub>2</sub> as well. Similar to Li (2005), Ding (2008a, b) claims that *yídìng*<sub>1</sub> expresses strong volition and *yídìng*<sub>2</sub> denotes emphasis on the truth of an inference/judgment. Ding's (2008a, b) conclusion is similar to Li (2005) and hence suffers from the same problems.

Chen (2011) is mostly on the grammaticalization of *yídìng*. As for the semantics of *yídìng*, he claims that *yídìng* expresses strong volition or stipulation/inference. Since Chen's (2011) conclusion is identical to Li (2005) and Ding (2008a, b), and therefore is vulnerable to the same problems.

Two major problems shared by Chen (2011), Ding (2008a, b) and Li (2005) are the following. First, they do not discuss whether it is possible to provide a unified semantics for *yídìng*, and second, they do not discuss how the contextual influence on the semantics of *yídìng* as demonstrated in the two conversations in (1) should be dealt with.

S. Li (2009), Wang (2007) and Zhou (2014) focus on *shìbì*. These three studies are also reviewed in chronological order. Wang (2007) is on the lexicalization of *shìbì*. This paper suggests that *shìbì* describes an inference made based on a current situation. S. Li (2009) is about the historical development of *shìbì*. This study states that *shìbì* expresses an inference that some situation is certain to take place in the future, based on the current status of some other situation. Zhou (2014) provides a relatively detailed discussion on the semantic features of *shìbì*, but basically says the same thing as S. Li (2009) and Wang (2007). While epistemic necessity is one of the readings expressed by *shìbì*, these studies cannot explain why *shìbì* is not good in (4a), which also has an epistemic necessity reading, and neither do they take the intensification reading, such as (3a), into consideration.

Since the above reviewed papers do not provide a comprehensive picture for the semantics of *yídìng* and *shìbì*, further study is called for so that the unanswered questions can be addressed.

## 3 Semantics of *Yídìng* and *Shìbì*

### 3.1 The Data

*Yídìng* can either present a proposition without a modal expression or one with a modal expression. The utterances of B and of B' in (1), and the sentence (2a) are typical examples where *yídìng* presents a proposition not containing a modal expression. (3a) is an example where *yídìng* presents a modal containing a modal expression. Either case, *yídìng* is ambiguous between a strong epistemic reading and an intensification reading. Let's look at a few more examples.

- (5) a. Lìsì yídìng zài jiā.  
Lìsì YÍDÌNG at home  
'It must be the case that Lisi is at home.'  
Or, 'Lisi is definite at home.'

- b. Wángwǔ yídìng yǐjīng xiěwán  
Wangwu YÍDÌNG already write.finish  
gōngkè le.  
homework Prc  
'It must be the case that Wangwu has already finished his homework.' Or,  
'Wangwu definitely has finished his homework.'
- c. Zài xià jǐ tiān dà yǔ,  
Again rain several day heavy rain  
zhèlǐ yídìng fānshēng tǔshīliú.  
here YÍDÌNG happen mud.slide  
'If it rains heavily a few more days, it must be the case that mud slide will happen here.' Or,  
'If it rains heavily a few more days, mud slide definitely will happen here.'
- (6) a. Zhàoliù yídìng huì qí jiāotàchē.  
Zhaoliu YÍDÌNG DYN ride bike  
'It must be the case that Zhaoliu can ride a bike.' Or,  
'Zhaoliu definitely can ride a bike.'
- b. Sūnqī yídìng děi dǎsǎo fángjiān le.  
Sunqi YÍDÌNG DEON clean room Prc  
'It must be the case that Sunqi has to clean his room.' Or,  
'Sunqi definitely has to clean his room.'

Some native speakers I consult point out to me that, standing alone, (6b) preferably has an intensification reading, rather than a strong epistemic necessity reading. However, if we provide a context for the sentence, the strong epistemic necessity reading can be brought out. For example,

- (7) Sūnqī yídìng děi dǎsǎo fángjiān le.  
Sunqi YÍDÌNG DEON clean room Prc  
Zhè shì wǒ-de tuīcè. Tā-de fùmǔ  
this be my conjecture his parents  
yǐjīng shòubǔliǎo le.  
already tolerate.not Prc  
'It must be the case that Sunqi has to clean his room. This is my guess. His parents cannot tolerate it anymore.'

So, can a unified semantics be proposed for *yídìng*? I believe so. The examples presented in this section and previous sections tell us that the semantics of *yídìng* contains two parts. The first part provides an epistemic necessity reading, just

like *must* in English. The other part provides an intensification reading.

If we put aside the contextual influence on the semantics of *yídìng* for the moment, the semantics of *yídìng* can be modeled using Kratzer's (2012[1981], 1991) semantics of modal expressions. See (8).

- (8) Modal semantics for *yídìng*  
Modal base: Epistemic  
Modal force: Necessity  
Ordering sources: (a) doxastic or stereotypical, (b) affirmative

In (8), the modal base, modal force and one of the ordering sources in (a) together are actually the typical semantics for an epistemic necessary modal expression. The new idea here is the second ordering source, the affirmative ordering source. von Stechow and Iatridou (2008) propose that weak necessity modals such as *should* in English need two ordering sources for their semantics. The idea of two ordering sources is adopted here.

What is an affirmative ordering source? An affirmative ordering source orders possible worlds in terms of the speaker's affirmativeness toward a proposition.  $\leq_A$  represents an affirmative ordering source. Then, the ordering of two possible worlds based on an affirmative ordering source is defined as below.

- (9)  $v, w$  are possible worlds.  $p$  is a proposition.  
 $w \leq_A v$  iff  $\{p: p \text{ is affirmed in } v\} \subseteq \{p: p \text{ is affirmed in } w\}$   
(cf. Kratzer 2012[1981]: 39)

How about *shìbì*? I show that *yídìng* and *shìbì* are interchangeable in some cases, but not in others. For the purpose of discussion, I repeat the relevant examples in (10).

- (10) a. Zhè-ge shíhòu, Xiǎomíng yídìng/  
This-CL time Xiaoming YÍDÌNG/  
\*shìbì zài jiā.  
\*SHÌBÌ at home  
'At this moment, it must be the case that Xiaoming is at home.'

- b. Hūn hòu, rúguǒ zhù Yīnní, wǒ  
 married after if live Indonesia I  
 \*yídìng/shìbì cídào gōngzuò.  
 \*YÍDÌNG/SHÌBÌ resign job  
 ‘After getting married, if we live in In-  
 donesia, I definitely have to quit my  
 job.’
- (11) a. Rúguǒ wǒ bù néng chōngfèn  
 If I not can sufficient  
 gōngyìng shìchāng dehuà, wǒ-de  
 provide market Prc my  
 gùkè shìbì/yídìng huì cóng  
 customer SHÌBÌ/YÍDÌNG will from  
 bié chù gòu huò.  
 other place purchase goods  
 ‘If I cannot provide sufficiently in the  
 market, my customers definitely pur-  
 chase goods from somewhere else.’
- b. Yào jiàngdī chéngběn, zhōngyóu  
 want decrease cost CPC  
 yídìng/shìbì yào zēng  
 YÍDÌNG/SHÌBÌ DEON increase  
 chǎn  
 production  
 ‘If it wants to decrease cost, CPC defi-  
 nitely has to increase production.’

In (10a), *yídìng* is good, but *shìbì* is not. 331 examples of *shìbì* are retrieved from the online version of the Sinica Corpus. Examining these examples carefully, I find that, whenever *shìbì* is used, additional information must be present so that the sentence with *shìbì* can be inferred. For example, in (10b), moving to Indonesia after getting married leads to the event that the speaker has to quit his/her current job. The same reasoning applies to (11a, b).

Therefore, the first difference between *yídìng* and *shìbì* is that the former does not need the context to explicitly provide information based on which the proposition presented by *yídìng* can be inferred, whereas the latter does. In (10a), *shìbì* is not good because of lack of such information.

What happens if another modal expression, other than *yídìng* and *shìbì*, occurs in the sentences, such as (11a, b)? In these cases, *yídìng* and *shìbì* are interchangeable, and they are ambiguous as discussed above.

So, what is the semantics of *shìbì* and how is it related to that of *yídìng*? (10) sheds some light on this question. Again, putting contextual influence

aside, I propose that the modal base of *shìbì* and the ordering source related to the modal base are both underspecified, while the affirmative ordering source is always there for *shìbì*. *Shìbì* cannot be used in (10a) because information required to infer the proposition presented by *shìbì* does not exist. The lack of such information makes it impossible to resolve the underspecified modal base (and the underspecified ordering source) of *shìbì*.

On the other hand, in (10b), if one moves out of town, then it is most likely required for him/her to quit his/her current job in town. That is, the relation between the two clauses in (10b) indicates a deontic reading and the underspecified modal base of *shìbì* is resolved to circumstantial and the ordering source is related to a physical law: if one is not at a place, he cannot hold a job at that place.<sup>3</sup>

In sum, putting contextual influence aside, I propose the following. *Yídìng* has an epistemic modal base and two ordering sources. One is doxastic or stereotypical and the other is affirmative. An affirmative ordering source orders possible worlds in terms of the degree of speaker’s affirmativeness concerning a proposition. *Shìbì* has an underspecified modal base and two ordering sources. One of the ordering sources is underspecified as well because it needs to be compatible with the modal base. The other is an affirmative one.

### 3.2 Dynamic Semantics for *Yídìng* and *Shìbì*

Although, in Section 3.1, semantics are proposed, along the lines of Kratzer (2012[1981], 1991), for *yídìng* and *shìbì*, Kratzer’s semantics of modality cannot take care of contextual influence, which is demonstrated in the two conversations in (1). There is no mechanism in Kratzer’s semantics of modality (and in truth-conditional semantics as well) to deal with contextual effects.

Instead, I would like to propose dynamic semantics (Groenendijk and Stokhof 1991, Chierchia 1995, etc.)<sup>4</sup> for *yídìng* and *shìbì* so that contextual effects can be taken care of. Yalcin (2007) discusses why sentences such as *suppose that it is raining but it might not be* is infelicitous. In order to take care of embedded epistemic modals, a clause embedded under *suppose* must be interpreted accord-

<sup>3</sup> Let’s not consider, for the moment, work at home through internet or other special situations.

<sup>4</sup> For an excellent introduction to dynamic (modal) logic, please refer to Section 3.2, Portner (2008).

ing to what the subject supposes. Hence, one version of Yalcin’s (2007) proposal is as follows:

- (12) a.  $S_{w,x}$  is defined as  $\{w': w' \text{ is compatible with what } x \text{ supposes in } w\}$   
 b.  $\|x \text{ suppose } \phi\|^{c,s,w} = \{w: S_x^w \subseteq \|\phi\|^{c,S_w,x,w'}\}$   
 c.  $\|\text{Suppose that it is raining but it might not be}\| = \forall w' \in S_{w,x}: \|\phi\|^{c,S_w,x,w'} \text{ is true} \wedge \exists w' \in S_{w,x}: \|\neg\phi\|^{c,S_w,x,w'} \text{ is true}$

(12c) is a contradiction because it is not plausible that  $S_{w,x}$  contains a possible world where  $\phi$  and  $\neg\phi$  are both true at the same time. Yalcin’s (2007) idea applies to *yídìng* and *shìbì* as well because of the infelicity of the following example:

- (13) *tiān zhème hēi, xiànzài yídìng/shìbì*  
 sky so dark now YÍDÌNG/SHÌBÌ  
*zài xiàyù. #dànshì, yě yǒu kěnéng*  
 Prg rain #but also have possibility  
*méiyǒu*  
 not  
 ‘It is so dark. Now, it must be the case that it is raining, #but it may not be.’

But, Yalcin’s (2007) idea alone is not adequate for *yídìng* and *shìbì* because they denote a ‘strong’ epistemic necessity reading, rather than simple epistemic necessity. Is it possible to incorporate the affirmative ordering source as defined in Section 3.1 into an information state, i.e. what Yalcin (2007) refers to as *s*? Veltman’s (1996) proposal can help us here.

In order to account for the semantics of *normally* and *presumably*, Veltman (1996) propose that an information state is a pair  $\sigma = \langle \varepsilon, s \rangle$ . *s* is a proposition and Yalcin’s (2007) *s* or  $S_{w,x}$  is one type of Veltman’s (1996) *s*.  $\varepsilon$  is an expectation pattern, i.e. an ordering of possible worlds, where  $w \leq_\varepsilon v$  iff every expectation which is met by *v* is also met by *w* (Veltman 1996: 13).

Combining Veltman (1996) and Yalcin (2007), I propose that for *yídìng* and *shìbì* the information state  $\sigma$  is also a pair and that  $\sigma = \langle \mathcal{A}, s \rangle$ . *s* is a proposition, as in Veltman (1996) and Yalcin (2007).  $\mathcal{A}$  is an affirmative ordering, where  $w \leq_{\mathcal{A}} v$  if and only if every proposition which is affirmed to be true in *v* is also affirmed to be true in *w*.

In addition, in order to account for the high degree of affirmativeness in the semantics of *yídìng* and *shìbì*, we define absolutely affirmative words as (14a). We also need to update the affirmative ordering with a proposition, so that the proposition is true in the worlds where more propositions are affirmed to be true, as defined in (14b):

- (14) a. Absolutely affirmative worlds (cf.  $n_{\langle \varepsilon, s \rangle}$  in Veltman 1996: 14)  
 $\text{Aff}_{\mathcal{A}} = \{w \in W: \forall v \in W, w \leq_{\mathcal{A}} v\}$ , where *W* is the set of all possible worlds.  
 b. Updating an affirmative ordering  
 $\mathcal{A} \bullet \phi = \{ \langle w, v \rangle: w \leq_{\mathcal{A}} v \text{ if } v \in \phi, \text{ then } w \in \phi \}$

(14a) says the following:  $\text{Aff}_{\mathcal{A}}$  is a set of possible worlds each of whose members has more propositions affirmed to be true than one of the other possible worlds in *W*.  $\text{Aff}_{\mathcal{A}}$  is referred to as the absolutely affirmative worlds because all the worlds in this set contain only propositions affirmed to be true.

(14b) is the definition of updating  $\mathcal{A}$  with  $\phi$ :  $\mathcal{A} \bullet \phi$  is a pair  $\langle w, v \rangle$ , where, if  $\phi$  is true in *v*, then  $\phi$  is also true in *w*, that is, the affirmative ordering takes  $\phi$  into consideration. In this way, we can relate a proposition  $\phi$  to an affirmative ordering  $\mathcal{A}$ .

- (15) a. strong epistemic necessity reading  
 $\sigma \|yídìng(\phi)\|^M$   
 $= \langle \mathcal{A} \bullet \phi, s \subseteq \phi \rangle$  if  $\text{Aff}_{\mathcal{A}} \cap \{w: \|\phi\|^{w,M} = 1\} \neq \emptyset$  and *s* represents the speaker’s knowledge in *w*; or  
 $=$  absurd state, otherwise  
 b. intensification reading  
 $\sigma \|yídìng(\phi)\|^M$   
 $= \sigma$  if  $\text{Aff}_{\mathcal{A}} \cap \{w: \|\phi\|^{w,M} = 1\} = \text{Aff}_{\mathcal{A}}$  and *s*  $\neq$  the speaker’s knowledge in *w*; or  
 $=$  absurd state, otherwise.

(15a) accounts for the strong epistemic necessity reading *yídìng* can denote. The ordering source  $\mathcal{A}$  is updated with the proposition  $\phi$ . This update relates  $\phi$  to the order  $\mathcal{A}$  so that the affirmative ordering takes  $\phi$  into consideration. Just like Yalcin (2007),  $s \subseteq \phi$  says that  $\phi$  is interpreted with respect to *s*, the speaker’s knowledge. There is a condition for the new information state  $\langle \mathcal{A} \bullet \phi, s \subseteq \phi \rangle$  to hold:

$\phi$  must be true in one of the absolutely affirmative worlds. This condition is stated as  $\text{Aff}_A \cap \{w: \|\phi\|^{w, M} = 1\} \neq \emptyset$ . If the condition does not hold, then  $A \bullet \phi$  fails and uttering the  $\|\text{yídìng}(\phi)\|^M$  produces an absurd state.

As for the intensification reading, since this is not an inference or judgment,  $s$  does not equal to the speaker's knowledge in  $w$ . Instead of updating the information state, an intensification reading simply performs a test, as stated in (15b). As long as  $\phi$  is true in all of the absolutely affirmative worlds, then  $\|\text{yídìng}(\phi)\|^M$  gives back the original information state. If the condition does not hold, then an absurd state is yielded.

How about *shìbì*? As pointed out in Section 3.1, the difference between *yídìng* and *shìbì* lies in that the modal base of *shìbì* is underspecified. If we examine the information state  $\sigma$  carefully, we can find that  $s$  in  $\sigma$  functions in a way similar to a modal base. Hence, I propose that the  $s$  in the information state for *shìbì* is underspecified and must be resolved before a sentence containing *shìbì* can get an appropriate interpretation. I formalize the idea as follows:

- (16) a.  $\langle A, s=? \rangle \|\text{shìbì}(\phi)\|^M$   
 b. Suppose that  $\alpha$ ,  $\phi$  forms a (mini) discourse.  $\alpha$ ,  $\phi$  are propositions  
 If  $\langle A, s=? \rangle$ ,  $\|\text{shìbì}(\phi)\|^M$  and  $R(\alpha, \phi)$ , then  $s = R$ .

In (16a),  $s = ?$  stands for an underspecified  $s$ . In (16b),  $R(\alpha, \phi)$  means that  $\alpha$  and  $\phi$  have a certain relation  $R$ . This  $R$  resolves the underspecified  $s$ . For example, in (10b), the two clauses are related because of a physical law, which says that one needs to live in a reasonable distance from where his job is. For this example, this physical law resolves  $s$  and hence (10b) can get an appropriate interpretation. Except for (16a, b), the semantics of *shìbì* is identical to that of *yídìng*, as in (15).

Now, with the dynamic semantics (15) and (16), we can successfully explain the two conversations in (1). For the conversation between A and B, since B says that this is a reasonable conjecture,  $s$  must represent the speaker's knowledge. Therefore, (15b) is ruled out. The information state is updated and we a strong epistemic necessity reading.

On the other hand, for the conversation between A and B', since B' says that this is a widely-known

fact,  $s$  cannot be equal to the speaker's knowledge. Hence, (15b) kicks in and we get an intensification reading.

In this section, I propose dynamic semantics for *yídìng* and *shìbì*. Both of these adverbials have an information state  $\langle A, s \rangle$ , where  $s$  is a proposition and  $A$  is an affirmative ordering. To derive a strong epistemic necessity reading, *yídìng* and *shìbì* update  $A$  with a proposition they present and specify that the proposition is a subset of or equal to  $s$ . This update holds if  $\phi$  is true in one of the absolutely affirmative worlds. To produce an intensification reading, a check is performed on an information state: if  $\phi$  is true in all of the absolutely affirmative worlds, the original information state is returned. If the condition is not satisfied, neither strong epistemic necessity reading nor intensification reading can be produced. This is the unified semantics for *yídìng* and *shìbì*.

Their difference is that the  $s$  in an information state  $\langle A, s \rangle$  for *shìbì* is underspecified, and needs to be contextually resolved so that a proposition presented by *shìbì* can get a proper reading.

## 4 Conclusion

In this paper, I propose dynamic semantics for *yídìng* and *shìbì* because truth-conditional semantics cannot deal with contextual effects in the semantics of *yídìng* and *shìbì*. Following Veltman (1996), I propose an information state  $\sigma$  is a pair  $\langle A, s \rangle$ , where  $s$  is a proposition and  $A$  is an affirmative ordering. *Yídìng*( $\phi$ ) performs update on an information state:  $A$  is update with  $\phi$  and  $s$  is specified to be a subset of or equal of  $\phi$ , as long as  $\phi$  is true in one of the absolutely affirmative worlds. Otherwise, uttering *yídìng*( $\phi$ ) leads to an absurd state. This is how a strong epistemic necessity reading is derived.

On the other hand, to yield an intensification reading, *yídìng*( $\phi$ ) performs a test on an information state. *Yídìng*( $\phi$ ) gives back the original information state as long as  $\phi$  is true in all of the absolutely affirmative worlds. Otherwise, an absurd state is produced.

As for *shìbì*, its semantics is identical to that of *yídìng*, except for the following: the  $s$  in an information state  $\sigma$  for *shìbì* is underspecified and needs to be resolved before a proposition presented by *shìbì* can get an appropriate interpretation. The

information needed to resolve the underspecified *s* for *shìbì* must be inferred from the context.

### Acknowledgements

I hereby acknowledge the financial support from Ministry of Science and Technology, Taiwan, under the grant number MOST 103-2410-H-194-037. I also thank my part-time research assistant Hsuan-Hsiang Wang for collecting data and for a preliminary analysis.

### References

- Chang, Yung-Li. 2000. Hànyǔ lùnduàn huì de yǔyì [On the Semantics of Predictive-Assertive *huì* in Mandarin]. Paper presented at the 9<sup>th</sup> International Conference on Chinese Linguistics. Singapore.
- Chen, Yong. 2011. Yíding de xūhuà jí liǎng zhǒng yǔyì de fēnhuà [Grammaticalization of *Yíding* and Two Types of Modality Diversification]. *Journal of Wuhan University of Science & Technology (Social Science Edition)*, 13, 5, pp. 605-609.
- Chierchia, Gennro. 1995. *Dynamics of Meaning: Anaphora, Presupposition and the Theory of Grammar*. Chicago: University of Chicago.
- Ding, Ping. 2008a. Yě shuō fùcí yíding [Adverb *Yíding* Revisited]. *Journal of Northwest University for Nationalities (Philosophy and Social Sciences)*, Year 2008, Issue 5, pp. 108-112.
- Ding, Ping. 2008b. Yíding yǔ kěndìng zuò zhuàngyǔ shí de bǐjiào [Comparison of Adverb *Yíding* and Adverb *Kěndìng*]. *Journal of Southwest University for Nationalities (Humanities and Social Sciences)*, Year 2008, Issue 8, pp. 236-240.
- von Fintel, Kai and Sabine Iatridou. 2008. How to Say *Ought* in Foreign: The Composition of Weak Necessity Modals. In *Time and Modality*. Eds. J. Guéron and J. Lecarme. Pp. 115-141. Berlin: Springer.
- Groenendijk, Jeroen and Martin Stokhof. 1991. Dynamic Predicate Logic. *Linguistics and Philosophy*, 14, pp. 39-100.
- Kratzer, Angelika. 1991. Modality. In *Semantics: An International Handbook of Contemporary Research*. Eds. von Stechow, A., Wunderlich, D. Pp. 639-650. Berlin: de Gruyter.
- Kratzer, Angelika. 2012[1981]. The Notional Category of modality. In *Modals and Conditionals*. Ed. Angelika Kratzer. Pp. 21-69. Oxford: Oxford University.
- Li, Chengjun. 2005. Fùcí yíding shuōluè [On Adverb *Yíding*]. *Lilù yùkǎn* [Theory Monthly], Year 2005, Issue 5, pp. 126-127.
- Li, Suying. 2009. Yǔqì fùcí shìbì de xíngchéng [On the Formation of Modal Adverb *Shìbì*]. *Yǔwén xuékǎn* [Journal of Language and Literature], Year 2009, Issue 10, pp. 42-44.
- Liu, Hsiao-mei. 1997. *Guó Mǐn Kèyǔ de dòngtài wénfǎ tǐxì jí dòngtài cí de shàngjiā dòngmào yǔyì* [Mood System and Interaction between Mood and Aspect in Mandarin, Taiwanese and Hakka]. Taipei: Crane.
- Portner, Paul. 2008. *Modality*. Cambridge: Cambridge University Press.
- Veltman, Frank. 1996. Defaults in Update Semantics. *Journal of Philosophical Logic*, 25, pp. 221-261.
- Wang, Meihua. 2007. Shìbì de cíhuì huà [Lexicalization of *Shìbì*]. *Journal of Hunan First Normal College*, 7, 1, pp. 101-103.
- Xu, Xiaomei. 1995. Qiǎn tán hànyǔ wùbì, yíding, quèxìn zài yīngyǔ zhōng de biǎodáfǎ [On the English Translations of *Wùbì*, *yíding* and *Quèxìn*]. *Huáiyīn gōngyè zhuānkē xuéxiào xuébào* [Journal of Huaiyin Junior College of Industry], 4, 1, pp. 56-57.
- Yalcin, Seth. 2007. Epistemic Modals. *Mind*, 116, pp. 983-1026.
- Zhou, Minli. 2014. Xīnwén bàodǎo yǔtǐ zhōng de fùcí shìbì qiǎnxī – jiān tán yǔ birán de bǐjiào [On Adverb *Shìbì* in News Report and Its Comparison with *Birán*]. *Journal of Xinyu University*, 19, 1, pp. 42-45.

## A Corpus-based Comparatively Study on the Semantic Features and Syntactic patterns of Yòu/Hái in Mandarin Chinese

**Yuncui Zhang**

National Language Resources Monitoring  
&Research Center, Beijing Language and  
Culture University

zycblcu@sina.com

**Pengyuan Liu\***

National Language Resources Monitoring  
&Research Center, Beijing Language and  
Culture University

liupengyuan@pku.edu.cn

### Abstract

This study points out that Yòu (又) and Hái (还) have their own prominent semantic features and syntactic patterns compared with each other. The differences reflect in the combination with verbs<sup>1</sup>. Hái (还) has absolute superiority in collocation with V+Bu (不)+V, which tends to express [durative]. Yòu (又) has advantages in collocations with V+Le (了)+V and derogatory verbs. Yòu (又)+V+Le (了)+V tends to express [repetition], and Yòu (又)+derogatory verbs tends to express [repetition, derogatory]. We also find that the two words represent different semantic features when they match with grammatical aspect markers Le (了), Zhe (着) and Guo (过). Different distributions have a close relation with their semantic features. This study is based on the investigation of the large-scale corpus and data statistics, applying methods of corpus linguistics, computational linguistics and semantic background model, etc. We also described and explained the language facts.

### 1 Introduction

The adverbs Yòu (又) and Hái (还) in Mandarin Chinese have a couple of meanings and usages respectively. For example, Lv Shuxiang (1983) Eight hundred words in Modern Chinese pointed out four kinds of moods and 13 kinds of semantic

items of Hái (还), about three kinds of semantic items and two kinds of moods of Yòu (又), such as [accumulation], [successive], and strengthen the negative mood, etc. Both of the two adverbs can be used to represent repetitive actions, the difference is that Hái (还) expresses imperfective actions and Yòu (又) expresses perfective actions. There has been abundant literature in studying the semantic features in linguistic field. To name a few, Hái (还) indicates [continuous] and Zài (再) expresses [repetition] (Jiang Qi and Jin Lixin, 1997). Ma Zhen (1999) differentiated the repetition semantic between Yòu (又), Zài (再) and Hái (还), Ma Zhen (2001) described the usage of Yòu (又) from a semantic background view, and she emphasized that it is essential to explain clearly the semantic background of every lexical or syntactic item that is taught. Gao (2002) studied the basic meaning of Hái (还) from a cognitive perspective. Zhang Yis (2004) regarded Yòu (又) and Hái (还) as time adverbs. According to Chen (1993), Yòu (又) emphasizes the [connection] from a perspective of discourse analysis, Hái (还) emphasizes different points in a sentence. Other scholars such as Chu (1983), Shi Xirao (1996), Shen (2001), Shi Jinsheng (2005), etc. Some researches above has indicated that adverbs Yòu (又), Hái (还) are easy to confuse under certain circumstances. Lots of scholars also summarized and compared their semantic features and syntactic behaviors, including their similarities and differences. But most of the studies applied introspective methods,

<sup>1</sup> Including verb grammatical structures in a general sense.

\* Contact Author.

which is subjective to a certain extent. Thus, some important issues have not arrived at any agreement yet.

These two adverbs can be used to represent complicated semantics, in which some semantics have a close contact. We think it is necessary to find out the prominent characteristic of each word, and then we can make comparative analysis effectively. This study applies the method of corpus linguistics and computational linguistics, combines with the comparison methods of function words and semantic background model, to verify with syntactic behaviors. Based on the statistics of a large number of language facts, we draw some conclusions which are supported by the data, and we also analyze the specific facts. Firstly, from calculating the collocation frequency with verbs, we get their own prominent semantic and syntactic features of each word. According to the analyses of the data, we found that: 1) Hái (还) tends to express [durative] and has superiority in collocation with V+Bu(不)+V. 2) Yòu (又) has obvious advantages in collocating with V+Le(了)+V and derogatory verbs. Pattern Yòu(又)+V+Le(了)+V means [repetition], Yòu(又)+derogatory verb means [repetition, derogatory]. In addition, the paper also studied the collocations with grammatical aspect markers Le(了), Zhe(着) and Guo(过). Yòu(又) is more easy to collocate with Le(了), Hái(还) has obvious advantages in matching Zhe(着) and Guo(过). Different collocations show different semantic features. The study also shows that it is the semantic features of the adverbs that cause the different choices of collocation with the aspect markers. This study is beneficial to the study of aspect functions of time adverbs and language learning to a certain extent. Corpus resources: BCC2 and DCC3. BCC is a large-scale comprehensive corpus, which contains microblog, science and technology, literature, the press, 15 billion words in total. DCC only contains newspaper corpora. Being the important part of written language, DCC will not make grammatical

mistakes. We selected 15 newspaper corpora, 8 billion words in total.

## 2 Collocations with Verbs and Semantic Features

### 2.1 Collocation Data

From observing the corpus, it is found that these two adverbs can occur with different syntactic patterns. Hái (还) tends to modify “vbv”, such as “Ni Hai Ji Bu Ji De wo” (你还记不记得我) [Do you still remember me], Zuo Bu Zuo Zhen (坐不坐诊) [be in clinic or not], Kao Bu Kao Shi (考不考试) [take an exam or not]. Yòu (又) tends to modify “vlv” and derogatory verbs, such as Ta Yòu Deng Le Deng (他等了等) [He waited a while again], Yòu Qiao Le Qiao (又敲了敲) [somebody knocked again], Ta You Tong Lou Zi (他又捅娄子) [He got into trouble again].

Table 1 below shows the frequency distributions and proportional distribution of Hái (还) and Yòu (又) in corpus.

	vbv freq	vlv freq	derogatory v freq
Hái	131	31	1346
Yòu	5	203	4445
total	136	234	5791

Table 1: The corpus distribution of Hái and Yòu

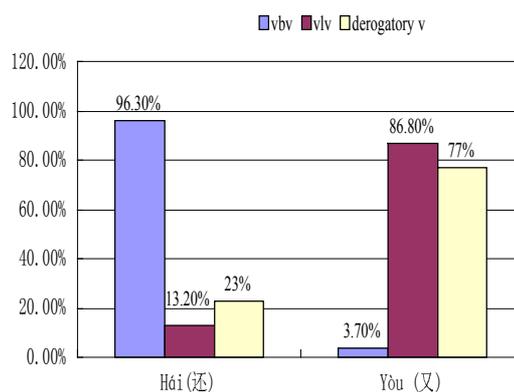


Figure 1: Proportional Distributions

<sup>2</sup> BCC (<http://bcc.blcu.edu.cn/>) was developed by Institute of Big Data and Educational Technology, Beijing Language and Culture University.

<sup>3</sup> DCC (<http://dcc.blcu.edu.cn/>) was developed by National Language Resources Monitoring & Research Center, Beijing Language and Culture University.

Table 2-4 includes top five syntactic patterns collocations of Hái (还) and Yòu (又). Data from high to low arrangement.

syntactic patterns	freq	syntactic patterns	freq
还/d 记不记得	40	又 /d 能	5
还/d 需不需要	18	不能/vbv	
还/d 值不值得	12		
还/d 愿不愿意	10		
还/d 应不应该	8		

Table 2: Hái/Yòu+vbv

syntactic patterns	freq	syntactic patterns	freq
又/d 缺乏/v	389	还/d 涉嫌/v	273
又/d 遭遇/v	366	还/d 乱/v	258
又/d 遭受/v	287	还/d 导致/v	231
又/d 遭到/v	235	还/d 威胁/v	152
又/d 涉嫌/v	230	还/d 遭到/v	130

Table 3: Hái (还)/Yòu (又)+derogatory verbs

syntactic patterns	freq	syntactic patterns	freq
又/d 看了看	74	还/d 拍了拍	7
又/d 指了指	26	还/d 看了看	6
又/d 想了想	20	还/d 指了指	4
又/d 摇了摇	14	还/d 摸了摸	3
又/d 摸了摸	11	还/d 挥了挥	1

Table 4: Hái/Yòu+vlv

By analyzing the collocation and the data, we found that the two words present a very different distribution. As shown above, Hái (还) collocates frequently with “vbv”. There are two kind of forms assume superiority in collocation with Yòu (又), one form is “vlv”, another is the derogatory verbs. We also found that the different syntax patterns carry different semantic features.

Table 5 below are the corresponding semantic features distributions of Hái (还) and Yòu (又). Both of the two words can express [additional], and also have their own prominent features.

	Hái (还)		Yòu (又)	
vbv	durative		additional	
	freq	%	freq	%
	131	100	5	100
vlv	additional		repetition	additional
	freq	%	freq	%
	31	100	193	95
derogatory v	additional		repetition, derogatory	
	freq	%	freq	%
	1346	100	4445	100

Table 5: The semantic distributions of Hái and Yòu From Table 5, we can see that all of the patterns Hái (还)+vbv 100% indicate [durative]. Both of the patterns Hái (还)+vlv and Hái (还)+derogatory verbs 100% indicate [additional].

Yòu (又)+vbv 100% expresses [additional], but only 5% of Yòu +vlv expresses [additional], 95% of Yòu (又)+vlv represents [repetition]. Yòu (又)+derogatory verbs 100% represents [repetition, derogatory].

Tong Xiaoe (2002) pointed out eight kinds of semantic items of Hái (还). The 2nd item is that Hái (还) can be used to represent that something stays or keeps the same and original state. One of the syntactic patterns is Hái (还)+vbv.

In this study we sum up this kind of semantic feature of Hái (还) as [durative]. In table 2, [durative] is the prominent semantic feature of Hái (还) compared with Yòu (又).

As we all know, in modern Chinese, Le (了) particle indicates that the completed action occurred in the past. In this type of sentence construction, and Le (了) can be marked for the perfective aspect according to previous studies. For instance, Wo Kan Le Kan (我看了看) [I just saw it for a while], Tui Le Tui (推了推) [gave a push]. Both of the two examples are completed actions in the past.

In table 5, the prominent semantic feature of Yòu (又) is completely different from Hái (还). Yòu (又) has its own prominent meaning--[repetition], and expresses a derogatory sense when modifying a derogatory verb. The sentence including Yòu (又) expresses something that is not looking forward to

according to Chu (1983:58), so from pattern Yòu (又)+derogatory verbs, we find the reason for Chu's study.

## 2.2 Description and Explanation

So far, we have verified the different semantic features and syntactic patterns between Hái (还) and Yòu (又) by way of the data analyses and the corpus. Below are adequate descriptions. All the examples the study collected blow are from BCC. In order to save space, we won't indicate the corpus source.

**[durative]: Hái (还) + V + Bu (不) + V**

- (1) Bu Zhi Dao Ni Hai Ji Bu Ji De Wo (不知道你还记不记得我)[ I don't know if you still remember me.]
- (2) Jia Ru Xia Yu, Wo Men Hai Qu Bu Qu Shi Chang? (假如下雨, 我们还去不去市场) [Supposing it rains, will we still go to the market?]

Hái (还) above means "still", "as before" in English. "Ji Bu Ji De" (记不记得) means "Shi Fou Ji De" (是否记得)[remember or not]. As for Tong Xiaoe (2002), the syntactic pattern Hái (还) +v + bu(不) + vp implies a former item. In sentence (1), [Ni Hai Ji Bu Ji De (你还记不记得)] implies the former semantic background--{Ni Yi Qian Ji De Wo (你以前记得我) [You remembered me before]}.

**[additional]: Hái (还) + V + Le(了) + V**

- (3) Di San Tian, Wo He Ba Ba Gao Bie, Ba Ba Shuo Le Xie Gu Li Wo De Hua, Hai Pai Le Pai Wo De Jian Bang, Jiu Wang Che Zhan Zou Qu Le. (第三天我和爸爸告别, 爸爸说了些鼓励我的话, 还拍了拍我的肩膀, 就往车站走去了)[On the third day, I said good-bye to my father. My father encouraged me, and patted my shoulder, then he went to the station.]
- (4) Li Wei Dong Yu Men De Zheng Le Zheng Li Mao, Hai Mo Le Mo Jia Hu Zi(李卫东郁闷地正了正礼帽, 还摸了摸胡子)[Li Wei Dong tidied his hat depressingly, and check false beard].

"Hái (还)" in sentence (3), (4) means "additional", "and". Before "Hai Pai Le Pai Wo De Jian Bang (还拍了拍我的肩膀) [and patted my shoulder]", Ba Ba Shuo Le Xie Gu Li Wo De Hua (爸爸说了

些鼓励我的话) [My father encouraged me]. In sentence (4), "Li Wei Dong De Zheng Le Zheng Li Mao, Hai Mo Le Mo Jia Hu Zi (李卫东正了正礼帽还摸了摸胡子) [Li Wei Dong tidied his hat, and check false beard]" express that "besides tidying his hat, Li Wei Dong also checking his false beard".

**[additional] : Hái (还) + Derogatory Verbs**

- (5) Ta De Tong Nian Bu Jin Jing Li Guo Che Huo, Ta Suo Zai Jia Xian Hai Zao Yu Di Zheng.(他的童年不仅经历过车祸, 他所在家乡还遭遇地震)[In his childhood, he lived through an automobile accident, otherwise, his hometown encountered earthquake.]

In sentence (5), "Zao Yu (遭遇) [encounter]" is a derogatory verb, which expresses derogatory sense. Hái (还) means "in addition to". The whole sentence represents his unfortunate childhood.

**[repetition]: Yòu (又)+Le(了) +V**

- (6) Ta You Qiao Le Qiao Men, Deng Le Yi Huier, Ke Hai Shi Mei Ren Kai Men . (他又敲了敲门, 等了一会儿, 可还是没人开门) [He knocked again and waited, but nobody opened the door].

Yòu(又) expresses "repetition", means "again" in example (6). Ta Yòu Qiao Le Qiao (他又敲了敲)[He knocked again] is an completed action.

**[repetition,derogatory]: Yòu (又)+derogatory verbs**

- (7) Ta Er zi Zai Xue Xiao You Tong Lou Zi Le. (他儿子在学校又捅娄子了) [His son got into trouble again in school].
- (8) Ru Gou Wo Men Zhe Yang Zuo, Qing Bie You Zhe Guai Guai Wo Men. (如果我们这样做, 请别又责怪我们) [If we do like this, don't blame us again].

Yòu modifies derogatory verbs in the examples above. Eg(7) and (8), You Tong Lou Zi Le. (又捅娄子了) [got into trouble again], You Zhe Guai Guai (别又责怪我们) [don't blame us again] mean "Zai Yi Ci (再一次)[once again]". Eg (7) is used for the past tense. Eg (8) is a negative sentence with Bie (别) [don't], which is not a completed action.

**[additional]: Yòu (又) + V + Bu(不) + V**

- (9) Hai Zi De Tong Nian Zhi You Yi Ci, Ni Yao Hai Zi De Hui Yi Li Zuo Shen Mo Meng, You Neng Bu Neng Bao Zheng Ta Men Jiang

Lai De Ri Zi Quan Shi Fan Hua Si Jin Ne.  
 (孩子的童年只有一次, 如何保证一个快乐的童年, 又能不能保证他们将来的日子全是繁花似锦呢) [Childhood is only once, how to promise an happy childhood, and how can you promise them a prosperous future].

Eg (9) expresses “In addition to promising an happy childhood, you have to think about how to promise children a prosperous”. Yòu (又) here means [additional], not “again”.

**[additional] : “Yòu (又)+ V + Le(了) + V”**

(10) Ling Dao Jian Le Lan Tang Shen Shi Huan Xi,  
 Dang Xia Jiu An Pai Gong Zuo, You Wen Le Wen Ta De Jia Ting Sheng Huo Qing Kuang  
 (领导见了蓝塘甚是欢喜, 当下就安排工作, 又问了问她的家庭生活情况) [The leader was so happy when he met Lan Tang, arranged work for her right away, and consulted her family life].

In sentence (10), the semantic background of “You Wen Le Wen Ta De Jia Ting Sheng Huo Qing Kuang (又问了问她的家庭生活情况)[and the leader consulted her family life]” is “the leader was happy”, then “the leader arranged work for”. Yòu (又) means [additional] here. Yòu (又) can be replaced by Hái (还).

To sum up, we draw the conclusion that each one of the two adverbs has its own prominent characteristic in semantic and syntactic compared with each other. The prominent semantic features and syntactic patterns of Hái (还) and Yòu (又) are as follows: Yòu has obvious advantages in matching v+Le(了)+v and derogatory verbs. The prominent semantic feature of Yòu (又) is [repetition]. Hái (还) has absolute superiority in collocation with v+ Bu(不) + v, the prominent of Hái (还) is [durative].

### 3 Collocation with Aspect Markers and Corresponding Semantic Features

#### 3.1 Collocation Data with Le, Zhe and Guo

According to Gong Qianyan (1994) and Ma Qingzhu(2004), time adverbs can represent abundant aspect and tense systems in Mandarin Chinese. The syntax pattern of aspect system in Mandarin Chinese is {Adverbs+V+Dynamic Auxiliary}. So adverbs have important aspectual functions. As grammatical aspect markers , there

are a series of detailed linguistic analyses on Le (了), Zhe (着) and Guo (过).

The particle Le (了) is generally considered a perfective marker, Guo (过) is considered as experiential aspect (Xiao and McEnery, 2004), and perfective aspect according to Chen Qianrui (2003). Zhe (着) is considered as the progressive aspect according to Chen Qianrui (2003), and durative aspect according to Xiao and McEnery(2004).

Through the statistical analysis in BCC, we found the regular collocations with grammatical aspect markers Le (了), Zhe (着) and Guo (过) between Yòu (又) and Hái (还).

	Le freq	Zhe freq	Guo freq
Hái	382395	70056	21660
Yòu	742970	0	5332
Total	1125365	70056	26992

Table 6: Collocation with Aspect Markers

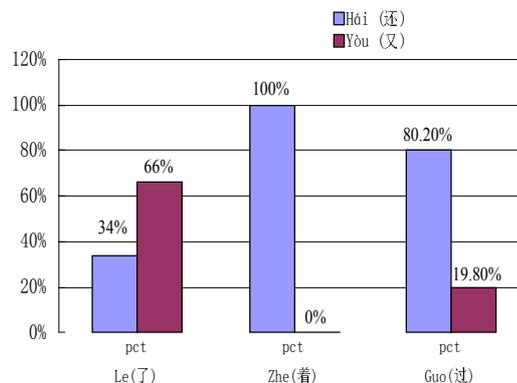


Figure 2: Proportional Distribution

From Table 6, in the case of Le (了), we can see Yòu (又)+Le (了) occupies the largest number, 742970 in total. That because Yòu (又) shows its own semantic feature [repetition], which is compatible with Le (了). For Zhe (着), obviously, only Hái (还) dominates the collocation.

From an overall perspective, we can see the different collocations with the grammatical aspect markers. Le (了) occupies the highest percentage with each adverb. Zhe (着) is inclined to match Hái (还), that is because Zhe (着) is a durative aspect

and [durative] is the exclusive feature of Hái (还). Due to the semantic compatibility of grammatically adjacent words, Hái (还) and Zhe (着) are compatible. The adverbs have different meanings when they collocate with aspect markers. From table 7 below, we can see that Hái (还) 100% expresses [additional] when collocating with Le (了) and Guo (过), which means “Er Qie (而且) [and]” and “Shen Zhi (甚至) [even]”; Yòu (又) 80% expresses [additional] collocating with Guo (过), 36% expresses [additional] and 64% expresses [repetition] when collocating with Le (了).

	Hái (还)		Yòu (又)			
le	additional		additional	repetition		
	num	%	num	%	num	%
	382395	100	2674	36	475501	64
zhe	durative		-			
	num	%	num	%		
	70056	100	0	0		
guo	additional		additional	repetition		
	num	%	num	%	num	%
	21660	100	4284	80	1048	20

Table 7: Semantic Features Distribution Collocation with Aspect Markers

Both Le (了) and Guo (过) indicate completed events or actions. When collocating with Zhe (着), Hái (还) 100% shows [durative]. Because Hái (还) is compatible with durative aspect Zhe (着). We will explain it adequately combining with example sentences below.

### 3.2 Description and Explanation

In chapter 3.1 we have list the collocations and semantic distributions with aspect markers Le (了), Zhe (着) and Guo (过) based on the data analyses and the corpus. Below are adequate descriptions and explanations. All the examples blow are from BCC. We will show the corresponding semantic features of every syntactic pattern.

**[additional]: Hái (还) + V + Le (了)/Guo (过)**

(11) Wo Gang Cai Shui Zhao Le, Hai Zuo Le Yi Ge Tian Mi De Meng. (我刚才睡着了, 还做了一个甜蜜的梦) [I fell asleep just now and had a sweet dream]

(12) Zhong Guo Xin Li Xue Bao Yi Ti Chang Ke Xue De Xin Li Xue Wei Zong Zhi , Nei Rong Yi Kan Deng Shi Yan Bao Gao Wei Zhu, Ci Wai Hai Kai Pi Le Xue Shu Tao Lu, Shu Bao Jie Shao, Xin Wen Bao Dao Deng Zhuan Lan. (《中国心理学报》以提倡科学的心理学为宗旨, 内容以刊登实验报告为主, 此外还开辟了学术讨论, 书报介绍, 新闻报道等专栏) [The Chinese Journal of Psychology aims to promote science, the content is given priority to with published experimental report. In addition, it also develops the academic discussion, books and newspapers as news column].

(13) Chu Li Yue Re Nei Lu Zhi Wai, Wo Hai Fan Wen Guo Ba Xi Di Yi Da Chen Shi Shen Bao Luo He Wei Yu Ya Ma Xun He Pan De Re Dai Yu Lin Chen Shi Ma Nao Si. (除里约热内卢之外, 我还访问过巴西第一大城圣保罗和位于亚马孙河畔的热带雨林城市玛瑙斯) [In addition to Rio DE Janeiro, Brazil, and I also visited Sao Paulo, the first city of Brazil , and Manaus located in the tropical forests of the amazon river].

(14) Zhong Guo De Nan Fan Ren He Bei Fang Ren Wo Dou Yan Jiu Guo, Wo Hai Zuo Guo Lei Si De Bao Gao Gei Ri Ben Zheng Fu. (中国的南方人和北方人我都研究过, 我还做过类似的报告给日本政府) [I studied People both from the south and the north of China, and I also made some similar reports for the Japanese government].

Sentence (11) means that Wo Bu Jin Shui Zhao Le, Er Qie Zuo Meng Le (我不仅睡着了, 而且做梦了) [I not only fall asleep, but also had a dream]. Sentence (12) means that Hai Kai Pi Le Xue Shu Tao Lu, Shu Bao Jie Shao, Xin Wen Bao Dao (还开辟了学术讨论、书报介绍、新闻报道等专栏) [it also develops the academic discussion, books and newspapers as news column] are based on Kan Deng Shi Yan Bao Gao (刊登实验报告) [ to publish experimental report], the latter is the semantic background of the form. And Hái (还) continuous use with “Ci Wai (此外)[in addition]”. Sentence (13), (14) is the same with (11) and (12). Hái (还) in eg (11), (12), (13), (14) express [additional].

**[durative]: Hái (还) + V + Zhe (着)**

(15) Dan You Yu Shou Ge Fang Mian Tiao Jian De Xian Zhi, Liang Di Min Jian Mao Yi Hai Cun

Zai Zhe Yi Xie Wen Ti (但由于受各方面条件的限制, 两地民间贸易也还存在着一些问题) [But due to the restriction from various aspects, there still exists some problems between the private trade].

- (16) Zhe Xie Dai Dai Xiang Chuan De Xi Su You Lai Yi Jiu. Gao Yuan Shang De Min Zu Chang Chang Chi Zhe Lei Si Ou Zhou Ren Zuo De Pi Sa Bin, Tong Shi Ta Men Hai Yi Zhi Bao Chi Zhe Yi Zhong Dui Niu De Chuan Tong Chong Bai (这些代代相传的习俗由来已久。高原上的民族常常吃着类似欧洲人做的比萨饼, 同时他们还一直保持着一种对牛的传统崇拜) [This custom has a long history carried on from generation to generation. The tribes on the plateau often eats pizza like Europeans did, at the same time they also still keep a kind of traditional worship to the cow].

Hái (还) + Zhe (着) in sentence (15) and (16) expresses [durative], which means “Reng Ran (仍然) [still]”, “Yi Zhi (一直) [all the time]”. And in sentence (16) Hái (还) continuously uses with “Yi Zhi (一直) [all the time]”. Hai Yi Zhi Bao Chi Zhe Yi Zhong Dui Niu De Chuan Tong Chong Bai (还一直保持着一种对牛的传统崇拜) [they also still kept a kind of traditional worship to the cow] equals to Yi Zhi Bao Chi Zhe (一直保持着)[kept all the time] or Hai Bao Chi Zhe (还保持着)[still kept]. Eg (15) is the same as eg (16).

**[additional]: Yòu (又) + V + Le (了)**

- (17) Mei Mei Tiao Le Yi Duo Fen Hong De Xiao Hua, You Dai Le Yi Dui Qing Qiao De Er Huan (妹妹挑了一朵粉红的小花, 又戴了一对轻巧的耳环) [The younger sister picked out a pink flower, and wore a pair of light earrings].

- (18) Zhe Shi Hou Qian Mian Cao Cong Li You Chu Xian Le Yi Ge Zhi Hui Guan. (这时候前面草丛里又出现了一个指挥官) [At this time, in front of the grass appeared another commander]

Eg (17) means that “the younger sister wore a pair of earrings besides picking out a pink flower”, not “wore again”. In eg (18), from “You Chu Xian Le Yi Ge Zhi Hui Guan (又出现了一个指挥官) [appeared another commander]”, we can get the semantic background “Zhi Qian Zai Cao Cong Li Chu Xian Guo Zhi Hui Guan (之前在草丛里出现过指挥官) [There was an commander in the grass

before]”, so Yòu (又) here means another Zhi Hui Guan (指挥官)[commander], which expresses [additional].

**[repetition]: Yòu (又) + V + Le (了)**

- (19) Wo Xiang Ting Ting Ta De Jian Yi, Suo Yi You Wen Le Liang Ge Guan Jian Xing De Wen Ti. (我想听听他的意见, 所以又问了两个关键性的问题) [I wanted to listen to his advice, so I asked two key questions again]

- (20) Zhong Guo Ao Yun Jian Er Zai Xi Ni You Qu De Le Li Shi Xing De Tu Po (中国奥运健儿在悉尼又取得了历史性的突破) [The Olympic athletes of China in Sydney made a historic breakthrough again]

Eg (19) and (20) are different from eg (17) and (18). Yòu (又) here indicates Zai (再) [again], which means [repetition]. Eg (19) means “I asked again”, and eg (20) means “The athletes made a breakthrough once more”.

**[additional]: Yòu (又) + V + Guo (过)**

- (21) Wen Yiduo Bu Dan You Shen Hou De Zhong Guo Wen Hua Su Yang, Tong Shi You Shou Guo 19 Shi Ji Lang Man Pai Chuan Tong He Wei Mei Zhu Yi De Ying Xiang. (闻一多不但有深厚的中国文化素养, 同时又受 19 世纪浪漫派传统和唯美主义的影响) [Wen Yiduo not only has profound Chinese culture accomplishment, but also influenced by the Romanticism tradition in 19th century and the influence of aestheticism].

Different from eg (19) or (20), eg (21) doesn't express “You Yi Ci (又一次) [once more/ again]”. The semantic background of “You Shou Guo Ying Xiang (又受过影响) [not only, ... but also influenced by ...]” is “Wen Yiduo You Shen Hou De Zhong Guo Wen Hua Su Yang (闻一多有深厚的中国文化素养) [Wen Yiduo has profound Chinese culture accomplishment]”. The whole sentence implies “Wen Yi Duo Shou Zhong Guo He Xi Fang Wen Hua De Ying Xiang. (闻一多受中国和西方文化的影响) [Wen Yiduo influenced by the Chinese culture and western culture]”. Yòu (又) here can be replaced by Hái (还), express [additional].

**[repetition]: Yòu (又) + V + Guo (过)**

- (22) Ta De Shang Kou You Lie Kai Guo (他的伤口又裂开过) [His body wound has cracked again].

(23) Lei Si De Yan Wu Zai Hai Sui Hou You Fa Sheng Guo Ji Ci, Dui Ying Guo Ren Chang Sheng Le Hen Da Zheng Dong. (类似的烟雾灾害随后又发生过几次, 对英国人产生了很大震动) [Similar smoke hazard then happened a few times, which have a great shock to the British]

Yòu (又) in (22), (23) express [repetition]. Yòu (又) here is equal to You/ Zai Yi Ci (又/再一次) [once again]. Sentence (22), the semantic background of “You Lie Kai Guo (又裂开过)[cracked again]” is “Shang Kou Yi Jin Li Kai Guo (伤口已经裂开过) [the wound had cracked once]”. The semantic background of “Yan Wu Zai Hai Sui Hou You Fa Sheng Guo Ji Ci (烟雾灾害又发生过几次) [smoke hazard then happened a few times]” is “Yan Wu Zai Hai Fa Sheng Guo (烟雾灾害发生过) [the smoke hazard happened once]”.

As a result, Yòu (又) and Hái (还) represent different semantic features when collocating with grammatical aspect markers Le (了), Zhe (着) and Guo (过). The different distributions of semantics and collocations have a close relation with their prominent semantic features.

#### 4 Conclusions

This study applies the method of corpus linguistics and computational linguistics, combines with the comparison methods and semantic background model. Based on the observation of a large number of data and analyses of the language facts, we point out that Yòu (又) and Hái (还) have their own prominent semantic features and syntactic patterns compared with each other. The differences also reflect in the combination with grammatical aspect markers. The aspect functions of Yòu (又) and Hái (还) have a close relation with their semantic features.

In this study, Hái (还) has absolute superiority in collocation with V + Bu (不) + V, which tends to express [durative]. Yòu (又) tends to express [repetition] when collocating with V+ Le (了) + V, express [repetition, derogatory] when collocating with derogatory verbs. The verbs modified by Yòu (又) express a derogatory sense.

We also studied the collocation with grammatical aspect markers Le (了), Zhe (着) and Guo (过).

Because syntactic performances are to do with their prominent semantic features, Yòu (又) is more easy to collocate with Le (了). Hái (还) has obvious advantages in matching Zhe (着) and Guo (过). Yòu (又)+V+Le (了) tends to indicate [repetition], and Hái (还)+Zhe (着) 100% express [durative]. This study is beneficial to the study of aspect functions of time adverbs and language learning to a certain extent. On the other hand, Chinese adverbs are more complicated than described, so we still need further study.

#### Acknowledgments

The study is supported by 1) National Language Committee Research Project (Grant No.WT125-45). 2) The Fundamental Research Funds for the Central Universities, and the Research Funds of Beijing Language and Culture University (No.15YCX101). 3) Science Foundation of Beijing Language and Culture University (supported by “the Fundamental Research Funds for the Central Universities”) (13ZDY03)

#### References

- Chen Qianrui. 2005. The contemporary aspect theory and the four levels aspect system of Chinese. *Chinese Language Journal*. (陈前瑞.当代体貌理论与汉语四层次的体貌系统. 汉语学报.2005.)
- Chen, Meiling . 1992. Discourse Functions of “Ye”, “You” and “Hai”. Unpublished Ntional Chengchi University paper.
- Chen, Meiling. 1993. ‘Ye’, ‘You’ and ‘Hai’ as Connectives in Chinese Narrative Discourse . Unpublished M. A. thesis, National Chengchi University.
- Chu, Chauncey C . 1998. A Discourse Grammar of Mandarin Chinese . New York: Peter Lang Publishing.
- Gao Zengxia. 2002. The basic semantic of adverb “Hai”, Chinese Teaching in the World. (高增霞, 副词“还”的基本义, 世界汉语教学, 2002)
- Gong Qianyan. 1994. The time system of Mandarin Chinese. *Chinese Teaching in the World*, 1994 (龚千炎.现代汉语的时间系统.世界汉语教学.1994)
- Guo Rui. 1993. The procedure structure of verbs in Modern Chinese, *Studies of The Chinese Language*, (郭锐, 汉语动词的过程结构, 中国语文, 1993(6))

- Lu Jianming and Ma Zhen. 1999. The theory of modern Chinese function words. Language Press, Beijing. (陆俭明,马真.现代汉语虚词散论.北京:语文出版社.1999.)
- Lu Shuxiang. 2013. Eight hundred words in Modern Chinese (Revised). Commercial Press, Beijing. 252-254, 633-634. (吕叔湘主编.现代汉语八百词(增订本).北京:商务印书馆.2013 :252-254, 633-634 ).
- Ma Qinzhu & Wang Hongbin. 2004. Time adverbs (previous / meanwhile/ afterwards) and verb category. International Conference Proceeding on aspect-tense system in Modern Chinese. BaiJia Press, Shanghai. (马庆株、王红斌.先时、同时、后时时间副词与动词的类[A].汉语时体系统国际研讨会论文集[C].上海:百家出版社,2004.)
- Ma Zhen. 2001. The modal adverbs “bing” and “you” strengthen the negative tone-concurrently talk about the words’ semantic background used in the sentences. Chinese Teaching in the World (马真.表加强否定语气的副词“并”和“又”—兼谈词语使用的语义背景,世界汉语教学,2001)
- Shen Jiakuan. 2001. The two syntactical structure related to “Hai”. Studies of The Chinese Language.(沈家煊 跟副词“还”有关的两个句式 中国语文,2001.)
- Shi Jingsheng. 2005. The dispute mood usage and grammaticalization of “You” and “Ye”. Chinese Teaching in the World. (史金生,“又”、“也”的辩驳语气用法及其语法化 世界汉语教学 2005)
- The Linguistics Institute of Chinese Academy of Social Sciences. 2014. The dictionary of Modern Chinese(sixth edition).Commercial Press, Beijing. (中国社会科学院语言研究所.现代汉语词典(第6版).北京:商务印刷馆,2014.)
- Tong Xiaoe. 2002. The develop of different semantic items of adverb “Hai”, Beijing Language and Culture University, Master’s Thesis. (童小娥.副词“还”的各项意义的演变.北京语言大学 硕士论,2002)
- Zhang Yisheng. 2000. The properties, scopes and categorization of Modern Chinese adverbs, Akademia Press, Shanghai. (张谊生,现代汉语的性质,范围和分类.上海:学林出版,2000).
- Zhang Yisheng. 2004. Modern Chinese adverb research. Akademia Press, Shanghai. (张谊生.现代汉语副词研究.上海:学林出版社,2004.)
- Zhu Dexi. 2011. Lecture Notes of Chinese Grammar. The Commercial Press,Beijing. (朱德熙,语法讲义.北京:商务印书馆,2011.)Lecture Notes of Chinese Grammar). The Commercial Press.

# An Empirical Study on Sentiment Classification of Chinese Review using Word Embedding

Yiou Lin    Hang Lei    Jia Wu    Xiaoyu Li

School of Information and Software Engineering  
University of Electronic Science and Technology of China  
Chengdu, China

lyoshiwo@gmail.com {hlei, jiawu, xiaoyu@uestc}@uestc.edu.cn

## Abstract

In this article, how word embeddings can be used as features in Chinese sentiment classification is presented. Firstly, a Chinese opinion corpus is built with a million comments from hotel review websites. Then the word embeddings which represent each comment are used as input in different machine learning methods for sentiment classification, including SVM, Logistic Regression, Convolutional Neural Network (CNN) and ensemble methods. These methods get better performance compared with N-gram models using Naive Bayes (NB) and Maximum Entropy (ME). Finally, a combination of machine learning methods is proposed which presents an outstanding performance in precision, recall and F1 score. After selecting the most useful methods to construct the combinational model and testing over the corpus, the final F1 score is 0.920.

## 1 Introduction

Sentiment analysis or opinion mining is the computational study of people’s opinions, appraisals, attitudes, and emotions toward entities, individuals, issues, events, topics and their attributes (Liu and Zhang, 2012). The task of sentiment analysis is technically challenging and practically very useful. For example, businesses always want to find public or consumer opinions about their products and services. Consumers also need a sounding board rather than thinking alone while making decisions. With the development of Internet, opinionated texts from social media (e.g., reviews, blogs

and micro-blogs) are used frequently for decision making, which makes automated sentiment analysis techniques more and more important. Among those tasks of the sentiment analysis, the key one is to classify the polarity of given texts. Many works have been done in recent years to improve English sentiment polarity classification. There are two categories of such works. One is called “machine learning” which is firstly proposed to determine whether a review is positive or negative by using three machine learning methods, including NB, ME and SVM (Pang et al., 2002). The other category called “semantic orientation” is applied to classify words into various classes by giving a score to each word to evaluate the strength of sentiment. And an overall score is calculated to assign the review to a specific class (Turney, 2002).

Recently, researchers have tried to handle tasks of Natural Language Processing (NLP) with the help of deep learning approaches. Among those approaches, a useful one called word2vec has attracted increasing interest. Word2vec translates words to vector representations (called word embeddings) efficiently by using skip-gram algorithm (Mikolov et al., 2013a). It is also proposed that the induced vector representations capture meaningful syntactic and semantic regularities, for example, “King” - “Man” + “Woman” results in a vector very close to “Queen” (Mikolov et al., 2013b).

Besides, with the advancement of information technology, for the first time in Chinese history, a huge volume of Chinese opinionated data recorded in digital form is ready for analysis. Though Chinese language plays an important role in economic

globalization, there are few works have been done for Chinese sentiment analysis with huge databases. It inspires us to make an empirical study on Chinese sentiment with bigger databases than usual.

The remain of the article is organized as follows: Section 2 briefly describes related work. Section 3 describes details of the methods used in training procedure. Section 4 reports and discusses the results. Finally, we summarize our works in Section 5.

## 2 Related work

According to Liu and Zhang (2012), the sentiment analysis research mainly started from early 2000 by Turney (2002) and Pang et al. (2002). Turney (2002) firstly used a few semantically words (e.g., excellent and poor) to label other phrases with the hit counts by queries through search engines. Then, researchers had also proposed several custom techniques specifically for sentiment classification, e.g., the score function based on words in positive and negative reviews (Dave et al., 2003) and feature weighting schemes used to enhance classification accuracy (Paltoglou and Thelwall, 2010). Besides, the other situation of sentiment analysis is to represent texts by vectors which indicate these words appear in the text but do not preserve word order. And a machine learning approach will be used for classification in the end. In such way, Pang et al. (2002) considered classifying documents according to standard machine learning techniques. In addition, subsequent research used more features in learning, making the main task of sentiment classification engineer an effective set of features (Pang and Lee, 2008).

However, compared to English sentiment analysis, there are relatively few investigations conducted on Chinese sentiment classification until 2005 (Ye et al., 2005). Li and Sun (2007) presented a study on comparison of different machine learning approaches under different text representation schemes and feature weighting schemes. They found that SVM achieved the best performance. After that, Tan and Zhang (2008) found 6,000 or bigger for the size of features would be sufficient for Chinese sentiment analysis, and sentiment classifiers were severely dependent on domains or topics.

Nowadays, inspired by the availability of large text corpus and the success of deep learning approaches, some researchers (e.g., Collobert et al. (2011), Johnson and Zhang (2014)) deviated from traditional methods and tried to train neural networks such as Convolutional Neural Networks (CNN) for NLP tasks (e.g., named entity recognition and sentiment analysis). Among them, Xu and Sarikaya (2013) and Kalchbrenner et al. (2014) got some state-of-the-art performance. But the work of Collobert et al. (2011) was paid most attention for describing a unified architecture for NLP tasks which learned features by training a deep neural network even when being given very limited prior knowledge. These NLP tasks included part-of-speech tagging, chunking, named-entity recognition, language model learning and semantic role labeling.

## 3 Methodology

This section presents the methodology used in our experiment.

### 3.1 Feature selection methods

#### 3.1.1 Sentiment lexicon and CHI

A sentiment lexicon accommodating sentiment words plays an important role in sentiment analysis. A combination of two Chinese sentiment lexicons (HowNet (Dong and Dong, 2006) and DLLEX (Xu et al., 2008)) is constructed, including 30406 words in total. After removing those words which do not appear in the corpus, 10444 sentiment words are preserved. After several experiments, CHI (Galavotti et al., 2000) is chosen for information gain. Finally, 150 most valuable words are added into the new lexicon. At last, 10543 words are obtained as features.

#### 3.1.2 Word2vec

Word2vec (Mikolov et al., 2013a) has gained kinds of traction today. As the name shows, it translates words to vectors called word embeddings. That is to say, it gets the vector representations of words. Gensim<sup>1</sup>, a python tool is used to get word2vec module. The method of training word2vec model is unsupervised learning and 300 is set as the quantity

<sup>1</sup><http://radimrehurek.com/gensim/>

of the dimension of vectors. Table 1 shows the word embeddings of a Chinese hotel review which means the room is very clean and neat. For convenient display, each value of dimension is multiplied by 10,000 and indicated by  $d_i$  ( $i = 1, \dots, 300$ ).

word	$d_1$	$d_2$	$d_2$	...	$d_{300}$
The room	-1102	-202	-668	...	-646
very	-6	355	-605	...	-460
clean	-287	-343	1077	...	-232
neat	-101	-399	-274	...	-986
average value	-374	-148	-118	...	-581

Table 1: An example of review vector

### 3.2 Traditional methods

#### 3.2.1 Naive Bayes Classification

Naive Bayes (NB) is widely used in sentiment classification which is used to classify a given review document  $d$  to the class  $c^* = \operatorname{argmax}_c P(c|d)$ . According to Bayes’s rule,

$$P(c_j|d) = \frac{P(c_j)P(d|c_j)}{P(d)}$$

where  $c_j$  is a kind of class and  $P(d)$  plays no role in selecting  $c^*$ . Let’s mark  $f_1, f_2, \dots, f_m$  as the set of features that appear in all reviews, and set  $n_i(d)$  as the number of times  $f_i$  appears in  $d$ . Usually,  $n_i(d)$  is set as 1, if  $f_i$  appears more than one time. Then, a formulation can be gotten as

$$P(c_j|d) = \frac{P(c_j) \prod_i^m P(f_i|c_j)^{n_i(d)}}{P(d)}$$

where the estimation of  $P(f_i|c_j)$  is calculated as follows, using add-one smoothing

$$\hat{P}(f_i|c_j) = \frac{1 + n_{ij}}{m + \sum_{k=1}^m n_{kj}}$$

#### 3.2.2 Maximum Entropy Classification

Maximum Entropy Classification follows the principle of maximum entropy (Jaynes, 1957), which means, subject to precisely stated prior data (such as a proposition that expresses testable information), the probability distribution which best represents the current state of knowledge is the one

with largest entropy. Thus, the estimate of  $P(c_j|d)$  is showed as follows

$$P(c_j|d) = \frac{1}{\pi(d)} \exp\left(\sum_{i=1}^m \lambda_{i,c_j} F_{i,c_j}(d, c_j)\right)$$

$$F_{i,c_j}(d, x) = \begin{cases} 1 & \text{if } n_i > 0 \text{ and } x = c_j \\ 0 & \text{otherwise} \end{cases}$$

where  $\pi(d)$  is a normalization function and  $\lambda_{i,c_j}$  is the weight of  $f_j$  in maximum entropy  $c_j$ . The other parameters are defined in the same way as Section 3.2.1. After fifteen iterations of the improved iterative scaling algorithm (Pietra et al., 1997) implemented in Natural Language Toolkit (Bird, 2006), the parameters of  $\lambda_{i,c_j}$  are adjusted to maximize the entropy of distribution of training data.

#### 3.2.3 Support Vector Machines

Support Vector Machines (SVM) is a very effective machine learning method firstly introduced by (Cortes and Vapnik, 1995). SVM constructs a hyperplane or a set of hyperplanes in a high dimensional space represented by  $\vec{w}$ . Since the larger the margin, the lower the error of the classifier, after training, the largest distance of support vector to nearest training-data point in any classes is achieved. Then the problem of maximizing the margin turns to

$$\operatorname{argmin}_{\vec{w}, b} \frac{1}{2} \|\vec{w}\|^2$$

where

$$y_i(\vec{w} \cdot x_i - b) \geq 1$$

and its unconstrained dual form is the following optimization problem: maximize  $\tilde{L}(\alpha)$  where

$$\begin{aligned} \tilde{L}(\alpha) &= \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j k(\mathbf{x}_i, \mathbf{x}_j) \\ &= \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j \mathbf{x}_i^T \mathbf{x}_j \end{aligned}$$

subject to  $\alpha_i \geq 0$  ( $i = 1, \dots, n$ ). Usually, the kernel here is linear, which means

$$k(\mathbf{x}_i, \mathbf{x}_j) = \mathbf{x}_i \cdot \mathbf{x}_j$$

For SVM models, python tool scikit-learn (Pedregosa et al., 2011) is chosen for training and testing. Scikit-learn<sup>2</sup> was started in 2007 as a Google

<sup>2</sup><http://scikit-learn.org>

Summer of Code project, and has become the most efficient and useful tool for data mining and analysis in Python. With all default parameters, LinearSVC and SVC with linear kernel are used in our article.

### 3.3 Ensemble methods

Ensemble methods (Dietterich, 2000; Friedman, 2001; Ridgeway, 2007) are supervised learning algorithm which commonly combine multiple hypotheses to form a better one. There are two families of ensemble methods, averaging methods and boosting methods. In averaging methods, several estimators will be built to average their predictions. It is a kind of vote, namely, on average. The combined estimator is usually better than any of the fundamental estimators since its variance is reduced (e.g., Bagging methods and Forests of randomized trees). By contrast, in boosting methods, fundamental estimators are built sequentially and each one tries to reduce the bias of the combined estimator. The idea behind it is to combine several weak models to generate a more powerful ensemble model (e.g., AdaBoost and Gradient Tree Boosting).

The ensemble method modules are chosen from scikit-learn, including AdaBoost, Gradient Tree Boosting and Random Forests. For each Chinese review, the average value of word embeddings is used as the input.

### 3.4 CNN methods

CNN is short for Convolutional Neural Networks. Its key module is to calculates the convolution between input and output. Just as CNN used in computer vision, a matrix is needed, as the input of CNN. After several experiments, we set  $D = 60$  as the dimension quantity of word embeddings for CNN. If there are  $L$  words in a sentence, combine their word embeddings together to construct a matrix of size  $L \times D$  as shown in Figure 1.  $L = 60$  is set since fixed  $L$  is needed, and which means, only 60 words are preserved from the beginning of a review. On the other hand, if the length is less than 60, the matrix will be filled with used vectors from the beginning of the review by repeating them. At last, every review is represented by a matrix of size  $60 \times 60$ .

Formally, in computer vision, given  $n$  images ( $X_l, l = 1, \dots, n$ ) of size  $r \times c$ ,  $k$  kernels of size

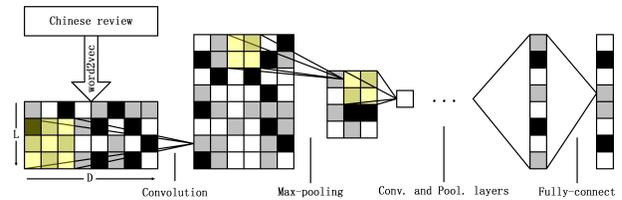


Figure 1: Illustration of Convnets

$a \times b$  are set. For each kernel patches a small image ( $X_s$ ) in the large image ( $X_l$ ),  $K(kernel_i, X_s)$  is computed, where  $K()$  is the kernel function, giving us  $k \times (r - a + 1) \times (c - b + 1)$  array of convolved features (more detail, see the tutorial<sup>3</sup>). Max-pooling is the key module to help training deeper model. It works like this: Expect that there is a  $60 \times 60$  matrix. Let's set pooling size  $10 \times 10$ , then the  $60 \times 60$  matrix will be divided into 36 small matrixes of size  $10 \times 10$ . Just pick the biggest value in each small matrix and combine them together. At last a  $6 \times 6$  matrix instead of  $60 \times 60$  matrix is gotten. Extending the implementation<sup>4</sup> of the lenet5 (LeCun et al., 1998), the convolutional layer and max-pooling layer are merged as one layer. The structure of ConvNets used is shown in Table 2.

Layer	Frame	Kernel	Kernel_size	Pool
1	$60 \times 60$	40	$5 \times 5$	$2 \times 1$
2	$28 \times 56$	50	$5 \times 5$	$2 \times 1$
3	$12 \times 52$	50	$5 \times 5$	$2 \times 1$
4	$4 \times 48$	—	—	—

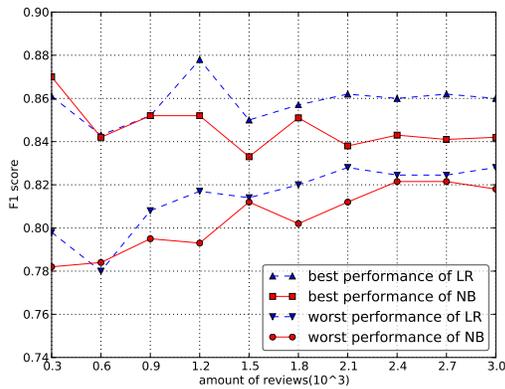
Table 2: Parameters of CNN layers

With the fourth layer, a fully-connect sigmoidal layer is constructed to classify the output values. After experiments, there are some rules can be concluded:

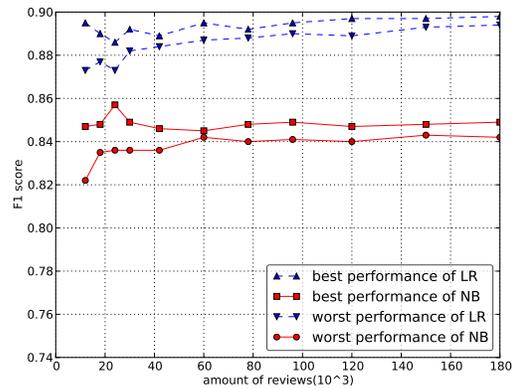
- The quantity of the word embedding dimensions shall be more than 50.
- Do not use pooling between the dimensions of word embedding (thus, in Table 2, the size of pooling is  $2 \times 1$ ).
- Adding more fully-connect sigmoidal layer dose not help in improving F1 score.

<sup>3</sup><http://ufldl.stanford.edu/wiki/index.php>

<sup>4</sup><http://deeplearning.net/tutorial/lenet.html#lenet>



(a) The worst performance of NB is worse than the best performance of LR



(b) The worst performance of NB is better than the best performance of LR

Figure 2: The performance curves with different amount of reviews.

## 4 Experiment results

### 4.1 Corporuses

Unlike English corporuses, Chinese corporuses are relatively small and usually focus on POS tagging (Mingqin et al., 2003), parsing (Xue et al., 2005) and translating (Xiao, 2010). In Chinese sentiment classification, the most popular corpus is ChnSentiCorp (Tan and Zhang, 2008) with 7,000 positive reviews and 3,000 negative reviews<sup>5</sup>. Since the amount of data collected by previous Chinese NLP researchers is too small for our work, we build a new corpus, MioChnCorp, with a million Chinese hotel reviews. The corpus is public and can be downloaded directly<sup>6</sup>. The reviews are crawled and filtrated from the website<sup>7</sup> which has coarse-grained rating (5-star scale) for each review. We give up the 3-star reviews which may be ambiguous, and mark the five-star and four-star reviews as positive and the rest as negative. Finally 908189 positive reviews and 101762 negative reviews are obtained. After word segmentation<sup>8</sup> being done, the sentiment classification process is executed.

Since ChnSentiCorp is small, the result may be unstable. Thus, Tan and Zhang (2008) gave the best performance and mean performance to evaluate a classification method. Zhai et al. (2011) tried to get

a believable result using the average value from 30 experiments. See Figure 2, Naive Bayes and Logistic Regression are used as classification methods to show the performance curves with different amount of reviews. The first sub-graph is tested on ChnSentiCorp, the second on is tested on MioChnCorp. Balanced corporuses are split into 3 equal-sized folds, two for training, the rest for testing. After repeating each experiment five times, the best performance and worst performance are marked. At last, two conclusions can be made: Firstly, when the amount of reviews is less than 60,000, the performance will be improbable (the best performance of model minus worst performance is bigger than 0.01). Secondly, more data usually help to get better performance, but the performance will be finally stable when data are big enough (e.g., 120,000 reviews).

### 4.2 The performance measure

F1 score (also called F-measure) is a measurement of a test’s accuracy which combines recall and precision as follows:

$$F_1 = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall}$$

$$Recall = \frac{\text{correct positive predictions amount}}{\text{positive example amount}}$$

$$Precision = \frac{\text{correct positive predictions amount}}{\text{positive predictions amount}}$$

since there are two categories (positive and negative) in MioChnCorp, Macro F1 is used to evaluate the

<sup>5</sup><http://www.datatang.com/data/11936>

<sup>6</sup><http://pan.baidu.com/s/1dDo9s8h>

<sup>7</sup><http://www.dianping.com/hotel>

<sup>8</sup><https://github.com/fxsjy/jieba>

performance of classification method over the corpus

$$\text{Macro F1} = \frac{\text{Positive F1} + \text{Negative F1}}{2}$$

in the rest of this article, F1 score means Macro F1 score.

### 4.3 Experimental design

Nine methods are designed to classify MioChnCorp using different features. NB and ME use 10543 words (the sentimental words and CHI words). LinearSVC use unigram and bigram. Five methods (SVC, LR, AdaBoost, Gradient Tree Boosting (GBT) and Random Forests (RF)) use the average vectors of word embeddings and CHI words (extending the dimension quantity to 450). CNN use the matrix constructed by word embeddings from words in a review as feature.

Though all of these models are effective, the combination of different machine learning methods is supposed to acquire better F1 score. There are two ways to combine those methods. First is vote, the idea is simple, “the minority is subordinate to the majority” (marked as Vote\_all ). The other way is to over-fit in the validate set. Add one more fold for validating into these tree folds. After training, nine models will be constructed. And each model gives one predication list for validating set. For each review, there are nine predications (e.g., [0 0 1 0 1 0 0 1 0]), 0 means negative, 1 means positive). Using the predication vectors of validating reviews as input, and the labels of validating reviews as the Logistic Regression output, after training on validating set, the combination model (called LR\_all) is built to test on testing set. The Framework of LR\_all is shown in Algorithm 1.

### 4.4 Comparison and analysis

Table 3, Table 4 and Table 5 show the performance of different machine learning methods. Subjected to hardware recourse (RAM:8G, CPU:Intel I5, GPU:GTX960M), the experiments are explored over corpuses with tree size: 40,000, 80,000 and 120,000. Each corpus is divided into four folds which are equal in size, two for training, one for validating, the rest for testing.

---

#### Algorithm 1 Framework of combination model

---

##### Input:

The experiment set of labelled samples:  
*train\_set*, *validate\_set* and *test\_set*;  
*n* machine learning classifiers (marked as  $C_i$ ,  $i = 1, \dots, n$ ) with default parameters

##### Output:

- For each classifier  $C_i$ , train on *train\_set*;
- 1: Test on *validate\_set* by  $C_i$  and storing predication as *validate\_list\_i*;
  - 2: Use logistic regression to predicate the label list of *validate\_set* by combination data of *validate\_list\_i* ( $i = 1, \dots, n$ ) and store the model as *LR\_all*;
  - 3: For each classifier  $C_i$ , test over test set, and storing the predication as *test\_list\_i*;
  - 4: Use *test\_list\_i* as input of *LR\_all* and produce final predication of *test\_set*, storing as *test\_list*
- 5: **return**  $C_i$  ( $i = 1, \dots, n$ ), *LR\_all*, *test\_list*;
- 

There are nine methods to construct the LR\_all model, but not all of them make contribution. Weka Explorer<sup>9</sup> provides attribute selection module to choose most useful attributes to the target attribute (namely, the label list of *validate\_set* in our situation). Extracting the *validate\_list\_i* ( $0 \leq i < n$ ) used in Algorithm 1, and combining these nine prediction lists with the label list of *validate\_set*, totally, ten attributes will be gotten. With 10-fold cross-validation, CfsSubsetEval attribute evaluator and BestFisrt search Method, Weka selects five most valuable attributes (ME, SVC, LinearSVC, RF and CNN). It is reasonable because they are most outstanding machine learning models which represent their own feature selection methods. Considering the limit of hardware resource and running time, LR is used to instead of SVC and CNN is abandoned. The result is shown in Figure 3. To our surprise, even only four feature is chosen, the F1 score is not reduced.

The more reviews we use in model building, typically the better performance we get till the performance is stable. SVM (linearSVC and SVC with linear kernel) has best performance not only

<sup>9</sup><http://www.cs.waikato.ac.nz/ml/weka/>

	Pre_0	Rec_0	Pre_1	Rec_1	F1
NB	.843	.896	.889	.833	.864
ME	.914	.850	.859	.910	.880
LinearSVC	.898	.881	.883	.900	.890
LR	.902	.879	.882	.905	.892
SVC	.910	.878	.882	.913	.895
Adaboost	.898	.867	.871	.902	.885
GBT	.897	.866	.870	.890	.883
RF	.910	.850	.860	.916	.883
CNN	.905	.865	.870	.909	.887
Vote_all	.790	<b>.955</b>	<b>.943</b>	.747	.849
LR_all	<b>.915</b>	.893	.896	<b>.917</b>	<b>.905</b>

Table 3: Different performance over 40,000 reviews

	Pre_0	Rec_0	Pre_1	Rec_1	F1
NB	.836	.900	.892	.823	.862
ME	.907	.853	.861	.913	.883
LinearSVC	.895	.886	.887	.897	.891
LR	.910	.876	.880	.913	.894
SVC	.910	.876	.880	<b>.914</b>	.895
Adaboost	.899	.866	.871	.903	.884
GBT	.897	.868	.872	.901	.885
RF	.910	.868	.874	<b>.914</b>	.891
CNN	.904	.864	.869	.909	.886
Vote_all	.786	<b>.957</b>	<b>.945</b>	.739	.846
LR_all	<b>.915</b>	.895	.897	.912	<b>.906</b>

Table 4: Different performance over 80,000 reviews

	Pre_0	Rec_0	Pre_1	Rec_1	F1
NB	.839	.900	.892	.827	.863
ME	.908	.850	.859	.913	.882
LinearSVC	.900	.891	.892	.901	.896
LR	.897	.882	.884	.900	.890
SVC	.905	.881	.884	.907	.894
Adaboost	.896	.864	.869	.899	.882
GBT	.896	.867	.871	.890	.883
RF	.910	.870	.876	.914	.892
CNN	.915	.853	.862	.920	.887
Vote_all	.777	<b>.965</b>	<b>.953</b>	.724	.842
LR_all	<b>.917</b>	.901	.903	<b>.919</b>	<b>.910</b>

Table 5: Different performance over 120,000 reviews

in traditional bag of words models, but also in word embedding models. Three ensemble methods work similarly and bigger data help to improve their performance obviously. There may be three reasons why CNN works better than NB and ME, but does

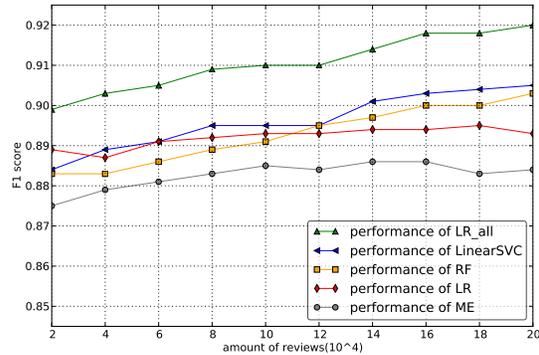


Figure 3: The performance curves of combination model and sub-models with different amount of reviews

not reach the expectant performance. Firstly, the amount of reviews is not big enough to train a deep learning model. Secondly, the architecture of the model may not be enough suitable as a language model. Finally, the features (word embeddings with 60 dimensions) for CNN is not accurate enough to present syntax and semantics in sentence. Vote\_all does not work well in improving performance, but has the highest negative recall and positive precision. LR\_all has better performance than Vote\_all because the same weights chosen by Vote\_all make these sub-models are equally important.

## 5 Conclusion and Future Work

In this article, an empirical study of sentiment categorization on Chinese hotel review is introduced. In order to conduct this experiment, a Chinese corpus, MioChnCorp<sup>10</sup>, with a million Chinese hotel reviews is collected. Using MioChnCorp, a word2vec model is trained to present distributed representations of words and phrases in Chinese hotel domain.

Then the experimental results indicate that the more data we use, the better performance we get. And 60,000 or larger size (e.g. 120,000) of reviews are sufficient in sentiment analysis of Chinese hotel review.

What’s more, we employ word embeddings as input features without any sentiment lexicons, and find such features perform well by using ensemble methods, LR, SVM and CNN. With respect to these learning methods, SVM works best. Though CNN

<sup>10</sup><http://pan.baidu.com/s/1dDo9s8h>

works not as good as expect, it still has better performance than NB and ME. The roles we used to construct the CNN model is introduce in Section 3.

Finally, a methodology, LR\_all is constructed to combine different machine learning methods and get an outstanding performance in precision, recall and F1 score of 0.920.

In the future, more work will be explored in building better CNN model for Chinese sentimental analysis and constructing combinational model in other tasks of NLP using word embedding.

## 6 Acknowledgements

The financial support for this work is provided by The Fundamental Research Funds for the Central Universities, No. ZYGX2014J065.

## References

- Steven Bird. 2006. Nltk: the natural language toolkit. In *Proceedings of the COLING/ACL on Interactive presentation sessions*, pages 69–72. Association for Computational Linguistics.
- Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. 2011. Natural language processing (almost) from scratch. *The Journal of Machine Learning Research*, 12:2493–2537.
- Corinna Cortes and Vladimir Vapnik. 1995. Support-vector networks. *Machine learning*, 20(3):273–297.
- Kushal Dave, Steve Lawrence, and David M Pennock. 2003. Mining the peanut gallery: Opinion extraction and semantic classification of product reviews. In *Proceedings of the 12th international conference on World Wide Web*, pages 519–528. ACM.
- Thomas G Dietterich. 2000. Ensemble methods in machine learning. In *Multiple classifier systems*, pages 1–15. Springer.
- Zhendong Dong and Qiang Dong. 2006. *HowNet and the Computation of Meaning*. World Scientific.
- Jerome H Friedman. 2001. Greedy function approximation: a gradient boosting machine. *Annals of statistics*, pages 1189–1232.
- Luigi Galavotti, Fabrizio Sebastiani, and Maria Simi. 2000. Feature selection and negative evidence in automated text categorization. In *Proceedings of KDD*.
- Edwin T Jaynes. 1957. Information theory and statistical mechanics. *Physical review*, 106(4):620.
- Rie Johnson and Tong Zhang. 2014. Effective use of word order for text categorization with convolutional neural networks. *arXiv preprint arXiv:1412.1058*.
- Nal Kalchbrenner, Edward Grefenstette, and Phil Blunsom. 2014. A convolutional neural network for modelling sentences. *arXiv preprint arXiv:1404.2188*.
- Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. 1998. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324.
- Jun Li and Maosong Sun. 2007. Experimental study on sentiment classification of chinese review using machine learning techniques. In *Natural Language Processing and Knowledge Engineering, 2007. NLP-KE 2007. International Conference on*, pages 393–400. IEEE.
- Bing Liu and Lei Zhang. 2012. A survey of opinion mining and sentiment analysis. In *Mining text data*, pages 415–463. Springer.
- Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013a. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.
- Tomas Mikolov, Wen-tau Yih, and Geoffrey Zweig. 2013b. Linguistic regularities in continuous space word representations. In *HLT-NAACL*, pages 746–751.
- Li Mingqin, Li Juanzi, Dong Zhendong, Wang Zuoying, and Lu Dajin. 2003. Building a large chinese corpus annotated with semantic dependency. In *Proceedings of the second SIGHAN workshop on Chinese language processing-Volume 17*, pages 84–91. Association for Computational Linguistics.
- Georgios Paltoglou and Mike Thelwall. 2010. A study of information retrieval weighting schemes for sentiment analysis. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 1386–1395. Association for Computational Linguistics.
- Bo Pang and Lillian Lee. 2008. Opinion mining and sentiment analysis. *Foundations and trends in information retrieval*, 2(1-2):1–135.
- Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan. 2002. Thumbs up?: sentiment classification using machine learning techniques. In *Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10*, pages 79–86. Association for Computational Linguistics.
- Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. 2011. Scikit-learn: Machine

- learning in python. *The Journal of Machine Learning Research*, 12:2825–2830.
- Stephen Della Pietra, Vincent Della Pietra, and John Lafferty. 1997. Inducing features of random fields. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 19(4):380–393.
- Greg Ridgeway. 2007. Generalized boosted models: A guide to the gbm package. *Update*, 1(1):2007.
- Songbo Tan and Jin Zhang. 2008. An empirical study of sentiment analysis for chinese documents. *Expert Systems with Applications*, 34(4):2622–2629.
- Peter D Turney. 2002. Thumbs up or thumbs down?: semantic orientation applied to unsupervised classification of reviews. In *Proceedings of the 40th annual meeting on association for computational linguistics*, pages 417–424. Association for Computational Linguistics.
- Richard Xiao. 2010. How different is translated chinese from native chinese? a corpus-based study of translation universals. *International Journal of Corpus Linguistics*, 15(1):5–35.
- Puyang Xu and Ruhi Sarikaya. 2013. Convolutional neural network based triangular crf for joint intent detection and slot filling. In *Automatic Speech Recognition and Understanding (ASRU), 2013 IEEE Workshop on*, pages 78–83. IEEE.
- LH Xu, HF Lin, and Jing Zhao. 2008. Construction and analysis of emotional corpus. *Journal of Chinese information processing*, 22(1):116–122.
- Naiwen Xue, Fei Xia, Fu-Dong Chiou, and Marta Palmer. 2005. The penn chinese treebank: Phrase structure annotation of a large corpus. *Natural language engineering*, 11(02):207–238.
- Qiang Ye, Bin Lin, and Yi-Jun Li. 2005. Sentiment classification for chinese reviews: A comparison between svm and semantic approaches. In *Machine Learning and Cybernetics, 2005. Proceedings of 2005 International Conference on*, volume 4, pages 2341–2346. IEEE.
- Zhongwu Zhai, Hua Xu, Bada Kang, and Peifa Jia. 2011. Exploiting effective features for chinese sentiment classification. *Expert Systems with Applications*, 38(8):9139–9146.

# Polarity Classification of Short Product Reviews via Multiple Cluster-based SVM Classifiers

Jiaying Song, Yu He, Guohong Fu

School of Computer Science and Technology, Heilongjiang University  
Harbin 150080, China

jy\_song@outlook.com, heyucs@yahoo.com, ghfu@hlju.edu.cn

## Abstract

While substantial studies have been achieved on sentiment analysis to date, it is still challenging to explore enough contextual information or specific cues for polarity classification of short text like online product reviews. In this work we explore review clustering and opinion paraphrasing to build multiple cluster-based classifiers for polarity classification of Chinese product reviews under the framework of support vector machines. We apply our approach to two corpora of product reviews in car and mobilephone domains. Our experimental results demonstrate that opinion clustering and paraphrasing are of great value to polarity classification.

## 1 Introduction

With the rapid development of social networks over the past years, sentiment analysis of short social media texts has been attracting an ever-increasing amount of attention from the natural language processing community (Hu *et al.*, 2004; Fu *et al.*, 2014; Santos and Gatti, 2014). While substantial studies have been achieved on sentiment analysis to date (Pang *et al.*, 2002; Hu *et al.*, 2004; Wang and Manning, 2012; Kim *et al.*, 2013; Liu *et al.*, 2014; He *et al.*, 2015), it is still challenging to explore enough contextual information or specific cues for polarity classification of short text like online product reviews (Fu *et al.*, 2014; Santos and Gatti, 2014). On the one hand, online product reviews are short and thus contain a limited amount of contextual information for sentiment analysis. On the other hand, online product reviews actually consist of opinions about a special product attributes. It is thus very difficult to capture a variety of attribute-specific cues in different product reviews for polarity

classification using a single general classifier. Furthermore, lacking large annotated corpora is still a fundamental issue for statistical sentiment analysis.

To address the above problems, in this work we explore review clustering and opinion paraphrasing to build multiple cluster-based classifiers for polarity classification of Chinese product reviews. To this end, we first explore a two-stage hierarchical clustering with multilevel similarity to cluster the training data into a set of opinion clustering and then building a polarity classifier for each review cluster via supported vector machines (SVMs). In addition, we also exploit paraphrase generation to expand product reviews in each cluster to achieve reliable training for the corresponding polarity classifier.

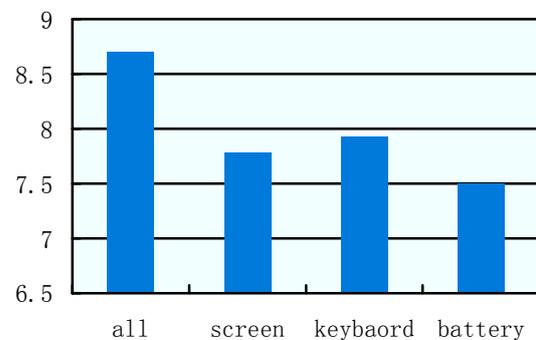


Figure 1. The entropy for product reviews in mobilephone domain before and after clustering.

Unlike most previous work with one classifier for polarity classification, our method uses multiple cluster-based classifiers to perform polarity classification, in which each classifier is tailored for a specific group of product reviews. At this point, our method actually provides a framework for attribute-based polarity classification and thus facilitate a feasible way to handle more attribute-specific cues for polarity classification. Therefore, we believe that cluster-based classification would be more precise in theory than most previous polarity

classification methods with a separate generic classifier. This hypothesis can be further demonstrated by Figure 1, which presents the entropy of the training data in mobilephone domain before and after clustering.

The rests of the paper proceed as follows. Section 2 provides a brief review of the literature on sentiment classification. Section 3 describes in details the proposed multiple cluster-based SVM classifiers for polarity classification of Chinese product reviews. Section 4 reports our experimental results on two sets of product reviews. Finally, section 5 concludes our work and discusses some possible directions for future research.

## 2 Related Work

Polarity classification is usually formulated as a binary classification problem (Turney, 2002; Pang and Lee, 2008). Most previous studies employ supervised machine learning methods to perform polarity classification on different linguistic levels such words, phrases, sentences and documents, including naïve Bayes model, support vector machines (SVMs), maximum entropy models (MEMs), conditional random fields (CRFs), fuzzy sets, and so forth (Pang *et al.*, 2002; Pang and Lee, 2008; Fu and Wang, 2010).

How to explore enough contextual information or specific cues is one important challenge for polarity classification of online product reviews (Fu *et al.*, 2014; Santos and Gatti, 2014). Actually, online product reviews are short text with a limited amount of contextual information for sentiment analysis. Furthermore, online product reviews actually consist of opinions about a special product attributes. It is thus very difficult to capture a variety of attribute-specific cues in different product reviews for polarity classification using a single general classifier.

Lacking large manually-annotated corpora is one of the major bottlenecks that supervised machine learning methods must face. To avoid this problem, some recent studies exploit bootstrapping or unsupervised techniques (Turney, 2002; Mihalcea *et al.*, 2007; Wilson *et al.*, 2009, Speriosu *et al.* 2011, Mehrotra *et al.* 2012; Volkova *et al.*, 2013). Unfortunately, unsupervised sentiment classifiers usually yield worse performance compared to the supervised counterparts.

Unlike most existing studies, in this study we attempt to build multiple cluster-based classifiers

for polarity classification of Chinese product reviews by exploring review clustering and opinion paraphrasing. We believe that our method can facilitate a feasible way to handle more attribute-specific cues for polarity classification of short product reviews on the web. Furthermore, to alleviate the problem of data sparseness, we further exploit paraphrase generation to expand training corpora for each review cluster. As such, our current study is also relevant to paraphrase recognition and generation. Although a variety of methods, from dictionary-based methods to data-driven methods (Madnani and Dorr, 2010; Zhao *et al.*, 2009), have been proposed for paraphrasing, here we do not want to look into paraphrasing issues. Instead, here we just employ the opinion element substitution based opinion paraphrase generation method (Fu *et al.*, 2014) to achieve enough data for training the proposed cluster-based polarity classifiers.

## 3 Our Method

In this section, we develop cluster based techniques to explore attribute-specific features for polarity classification of short product reviews.

### 3.1 Overview

As shown in Figure 2, our method involves two major processes, namely the SVM modeling process based on review clusters and the polarity classification process with the cluster-based SVM classifiers.

**Cluster-based SVM Modeling.** As can be seen in Figure 2, we divide the training process into three main steps: (1) In the review clustering step, we first cluster reviews in the training corpus into a set of clusters  $C=\{C_1, C_2, \dots, C_n\}$  in terms of product attributes; (2) In order to achieve enough data for reliable modeling for each cluster, in the second step we further expand the training set for each cluster  $C_i$  ( $1\leq i\leq n$ ) via opinion paraphrase generation and thus obtain sets of expanded training data  $EC_1, EC_2, \dots, EC_n$  for opinion clusters  $C_1, C_2, \dots, C_n$ , respectively; (3) We finally employ SVMs to build a classification model  $M_i$  for each cluster  $C_i\in C$  from the relevant expanded training data set  $EC_i$ . It should be noted that we have a special cluster  $C_x$  for all reviews that are out of any cluster in  $C$  during review clustering. For convenient, we refer to  $C_x$  as miscellaneous cluster and the relevant classification model (viz.  $M_x$ ) as miscellaneous model.

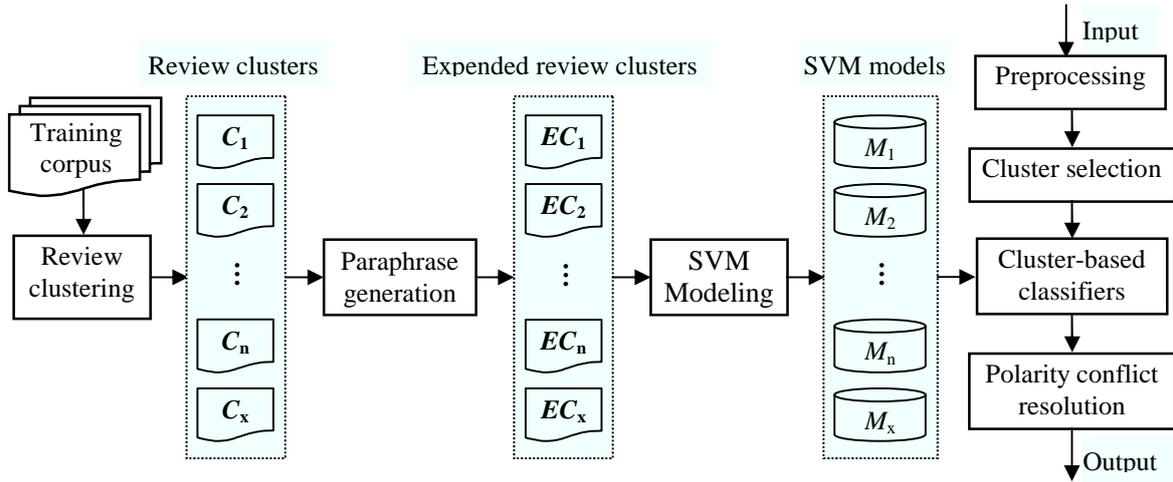


Figure 2. Overview of the cluster-based sentiment polarity classification system.

**Cluster-based polarity classification.** Given a input product review or opinionated sentence, we take four steps to determine its polarity category: To acquire linguistic information for subsequent polarity classification, in the preprocessing module we apply the morpheme-based lexical analyzer (Fu *et al.*, 2008) and the CRFs labeling technique to perform lexical analysis (viz. word segmentation and part-of-speech tagging) and opinion element recognition over the input, respectively. Then, we determine what clusters that the input should belong to in terms of the product attributes it contains. Thirdly, we employ the relevant cluster-based SVM classifiers to perform polarity classification. However, this step may yield different polarity classes for the input with multiple product attributes. So we finally use a polarity conflict resolution module to choose a final polarity for the input via a rule-based voting method.

### 3.2 Product Review Clustering

We cluster product reviews in the training data in terms of product attributes they contain. So the key to this task is how to resolve co-referred product attributes and implicit attributes in product reviews. To approach this, in this work we employ a two-stage hierarchically clustering algorithm with multilevel similarity.

#### 2.2.1 Similarity for explicit attribute clustering

In order to handle different levels of connections between explicit attributes in real product reviews, we consider two similarities, namely the literal similarity based on Jaccard coefficient, the word embedding based semantic similarity.

**Literal Similarity.** Literal similarity is used to handle the literal linking between co-referred

product attributes. Considering that edit distance cannot objectively reflect the real similarity for some co-referred feature expressions like 油耗 you-hao ‘fuel consumption’ and 耗油 hao-you ‘fuel consumption’, we exploit Jaccard coefficient in Equation (1) to calculate the literal similarity of two attributes  $a_1$  and  $a_2$ .

$$S_L(a_1, a_2) = \frac{|set(a_1) \cap set(a_2)|}{|set(a_1) \cup set(a_2)|} \quad (1)$$

Where,  $set(a_1)$  and  $set(a_2)$  denote the set of characters within  $a_1$  and  $a_2$ , respectively.

**Semantic Similarity.** In addition literal similarity, we also compute semantic similarity for some co-referred attributes without explicit literal connections, such as 像素 xiang-su ‘pixel’ and 分辨率 fen-bian-lv ‘resolution’. In order to avoid data sparseness, we use word embeddings (Mikolov, et al., 2013) to represent the semantics of product attributes. Given a pair of product attributes  $a_1$  and  $a_2$ , let  $vec(a_1)$  and  $vec(a_2)$  be their respective word embeddings. In order to map the cosine value to  $[0, 1]$ , then their similarity based on word embeddings, denoted by  $S_S(a_1, a_2)$ , can be defined by Equation (2).

$$S_C(a_1, a_2) = 0.5 + 0.5 \times \frac{vec(a_1) \bullet vec(a_2)}{|vec(a_1)| \times |vec(a_2)|} \quad (2)$$

Some complicated co-referred attributes may have both literal and semantic connections. To handle this problem, we further combine the above two similarity via linear interpolation and obtain the total similarity of a given explicit attribute pair, as shown in Equation (3).

$$S_{EA}(a_1, a_2) = \alpha \times S_L(a_1, a_2) + (1 - \alpha) \times S_S(a_1, a_2) \quad (3)$$

Where,  $\alpha$  is the interpolation coefficient.

2.2.2 Similarity for implicit attribute clustering

On the basis of the hypothesis that co-referred attributes tend to be collocated with similar evaluations, we thus exploit evaluation similarity to cluster reviews with implicit attributes. In particular, we consider explanatory evaluations as the context for implicit attributes because compared to non-explanatory evaluations, explanatory evaluations are feature-specific indicators for product attribute clustering (Kim *et al.*, 2013; He *et al.*, 2015), as illustrated by Table 1. To extract explanatory evaluations for implicit attribute clustering, we use the explanatory segment labeling technique by (He *et al.*, 2015).

Definitions	Examples
<b>A non-explanatory evaluation</b> only presents the sentiment orientation on a given target without any explanations for the reasons of the sentiment.	这个手机的屏幕还不错。 ‘The screen of this handphone is good.’
<b>An explanatory evaluation</b> not only presents the sentiment orientation on a given target but also explains the reasons of the sentiment.	这个手机的屏幕分辨率很高。 ‘The screen resolution of this handphone is very high.’

Table 1. Explanatory vs. non-explanatory evaluations in Chinese product reviews.

Let  $e_1$  and  $e_2$  be the respective explanatory evaluations for two implicit product attributes  $a_1$  and  $a_2$ ,  $Set(e_1)$  and  $Set(e_2)$  be the respective synsets of the explanatory keywords within  $e_1$  and  $e_2$ , we can then compute their evaluation similarity  $S_{IA}$  with Equation (4).

$$S_{IA}(a_1, a_2) = |Set(e_1) \cap Set(e_2)| / |Set(e_1) \cup Set(e_2)| \quad (4)$$

Here, we employ tf-itf to extract the explanatory keywords from the explanatory evaluations  $e_1$  and  $e_2$ , and then obtain their respective synsets from the training data for word embeddings via semantic paraphrasing (Bhagat and Hovy, 2013).

2.2.3 The two-stage clustering algorithm

In this work we use a two-stage hierarchical clustering algorithm to perform review clustering, as shown in Figure 3. Where,  $ClusterSimE(C_i, C_j)$  is the average similarity between each pair of explicit attributes from  $C_i$  and  $C_j$ , respectively, and  $ClusterSimI(r_i, C_j)$  is the average evaluation similarity between the evaluation in  $r_i$  and the evaluation within reviews from  $C_j$ .

- 
- Input:** A set of product reviews  $R = \{r_1, r_2, \dots, r_n\}$   
**Output:** A set of review clusters  $C = \{C_1, C_2, \dots, C_k\}$ .
1. Initialization: Separate  $R$  into two groups, namely the group  $R_E$  with explicit attributes and the group  $R_I$  with implicit attributes.
  - Stage 1:** clustering reviews with explicit attributes
    2. Let each review  $r_i \in R_E$  be a cluster  $C_i$  ( $1 \leq i \leq |R_E|$ ), and add it to  $C$ .
    3. For  $C_i \in C$ , if  $\exists C_j$  that makes  $ClusterSimE(C_i, C_j)$  be the maximum, and  $ClusterSimE(C_i, C_j) > \theta$ ,
    4. then merge clusters  $C_i$  and  $C_j$ , and update  $C$ .
    5. Repeat 2-4 until the number of clusters in  $C$  remains unchanged.
  - Stage 2:** clustering reviews with implicit attributes
    6. For each review  $r_i \in R_I$
    7. if  $\exists C_j \in C$  that makes  $ClusterSimI(r_i, C_j)$  be the maximum,
    8. then add  $r_i$  into  $C_j$ .
    9. Output  $C$  as the review clusters.
- 

Figure 3. The two-stage algorithm for Chinese product review clustering.

3.3 Opinion Paraphrase Generation

As we have mentioned above, the original training corpus will be separated into review clusters during review clustering. Each review cluster contains a group of reviews about a specific product attribute and are further used to training the specific classifier for the corresponding cluster. As a consequence, the dataset for some clusters may be too small for reliable training. To avoid this problem, we expand the review cluster via by paraphrasing each review via opinion element substitution (Fu *et al.*, 2014), which takes the following two main steps to generate all proper paraphrases for a given review  $R$ .

Items	Examples
Attribute	价格 ‘price’
Attribute co-references	价 价格 价钱 价位 ...
Positive evaluations	Low: 合适 适中 实惠 优惠 不高 公道 比较便宜 有优势 值 ...
Negative evaluations	High: 高 太高 真高 偏高 有点高 贵 太贵 偏贵 有点贵 不合理 有点无语 ...

Table 2. An example of equivalent attribute-evaluation pairs from the training data.

(1) **Opinion element substitution.** We first generate a set of potential paraphrases for  $R$  by substituting opinion elements, viz. the attribution and its evaluation in  $R$  with their equivalent counterparts extracted from the training corpus (as shown in Table 2), and then store them with

word lattice. For convenience, here we refer this word lattice as paraphrase word lattice.

(2) **n-best paraphrase decoding.** The generated paraphrase word lattice actually contains all potential paraphrases, including both proper and improper paraphrases for the input review  $R$ . To exclude the improper paraphrase candidates, we further employ bigram language models to decode  $n$ -best paths from the paraphrase word lattice, where each path forms a probable paraphrase for  $R$ .

### 3.4 Polarity Conflict Resolution

Polarity conflict will arise if the input review sentence receives multiple but different polarity classes after polarity classification. The reason may be due to the fact that an opinionated sentence in product reviews may have more than one attribution. In this case, the system will assign more than one cluster to the input during cluster selection, and further exploit multiple different classifiers to perform polarity classification. As a consequence, an input opinionated sentence may get different polarity categories after polarity classification. In this case, polarity conflicts will arise.

In order to avoid the potential polarity conflicts, we further employ a simple rule-based voting mechanism. Given a review sentence, let  $K_{POS}$  and  $K_{NEG}$  be the respective total number of positive classes and negative classes produced by the system. Thus, we can determine its final sentiment polarity using the following three rules.

- Rule 1. if  $K_{POS} > K_{NEG}$ , then the final polarity is positive.
- Rule 2. if  $K_{POS} < K_{NEG}$ , then the final polarity is negative.
- Rule 3. if  $K_{POS} = K_{NEG}$ , then the final polarity is the same as the one yielded by the miscellaneous classification model  $M_x$ .

## 4 Experimental Results and Analysis

To assess our approach, we have conducted experiments over two corpora of product reviews from car and mobilephone domains, respectively. This section reports our experimental results.

### 4.1 Experiment Setup

**Corpora.** We use two corpora of product reviews in car and mobilephone domains that are manually annotated with multiple levels of linguistic and sentiment information, including

word segmentation, part-of-speech tags, opinion elements and polarity classes. We further separate them into training and test sets, respectively. Table 3 presents the basic statistics of the experimental datasets.

Datasets	Car			Mobilephone		
	Total	Pos	Neg	Total	Pos	Neg
Training	1424	712	712	1266	633	633
Test	714	454	260	630	402	228

Table 3. Basic statistics of the experimental data.

**Sentiment Lexicon.** We use a sentiment lexicon in our system that contains a total of about 18K sentiment words built from the CUHK and NTU sentiment lexica<sup>1</sup> and HowNet<sup>2</sup>.

**Evaluation Metrics.** We employ macro average precision/recall/F-score (denoted by  $P_{macro}$ ,  $R_{macro}$  and  $F_{macro}$ , respectively) and micro average F-score (denoted by  $F_{micro}$ ) to evaluate polarity classification performance.

**LibSVM & Features.** Considering the focus of our current work, we employ the LibSVM toolkit (Chang and Lin, 2011) with a linear kernel and the traditional one-hot feature representation to build our system.

**Word embeddings learning.** To achieve word embeddings based semantic similarity for review clustering, the Google open source tool<sup>3</sup>, viz. word2vec, is used here to learn word embeddings from two larger corpora of car reviews (about 250K reviews) and mobilephone reviews (about 250K reviews). The dimension size is set to 100.

### 4.2 Effects of Different Parameters

As we have mentioned above, our clustering algorithm involves two parameters, viz  $\alpha$  and  $\theta$  for optimization. Where,  $\alpha$  determines the importance of the two similarity, namely the literal similarity and the semantic similarity, while  $\theta$  determines whether the clustering criteria is lenient or strict. In this work we employ the grid search (Bergstra and Bengio, 2012) to perform parameter optimization. Thus, we have  $\alpha=0.8$  and  $\theta=0.15$  for the mobilephone dataset, and  $\alpha=0.6$  and  $\theta=0.3$  for the car domain.

<sup>1</sup> <http://www.datatang.com/data/43460>

<sup>2</sup> <http://www.keenage.com/>

<sup>3</sup> <http://code.google.com/p/word2vec/>

$\theta$	$\alpha$	Mobilephone				Car			
		P <sub>macro</sub>	R <sub>macro</sub>	F <sub>macro</sub>	F <sub>micro</sub>	P <sub>macro</sub>	R <sub>macro</sub>	F <sub>macro</sub>	F <sub>micro</sub>
0.15	0.6	0.811	0.818	0.814	0.828	0.730	0.749	0.739	0.744
	0.7	0.845	0.846	0.846	0.859	0.782	0.776	0.779	0.800
	0.8	<b>0.856</b>	<b>0.874</b>	<b>0.865</b>	<b>0.871</b>	0.787	0.781	0.784	0.804
	0.9	0.849	0.859	0.854	0.864	0.790	0.796	0.793	0.809
0.20	0.6	0.830	0.841	0.835	0.846	0.770	0.750	0.760	0.785
	0.7	0.851	0.857	0.854	0.866	0.720	0.739	0.729	0.734
	0.8	0.840	0.851	0.846	0.856	0.797	0.787	0.792	0.812
	0.9	0.836	0.852	0.844	0.852	0.804	0.810	0.807	0.822
0.25	0.6	0.827	0.841	0.834	0.843	0.716	0.733	0.724	0.731
	0.7	0.837	0.854	0.845	0.853	0.802	0.812	0.807	0.820
	0.8	0.840	0.852	0.846	0.856	0.803	0.815	0.809	0.822
	0.9	0.831	0.850	0.840	0.847	0.787	0.794	0.790	0.806
0.30	0.6	0.830	0.841	0.835	0.846	<b>0.827</b>	<b>0.813</b>	<b>0.820</b>	<b>0.838</b>
	0.7	0.843	0.859	0.851	0.859	0.804	0.814	0.809	0.822
	0.8	0.836	0.854	0.845	0.852	0.797	0.805	0.801	0.816
	0.9	0.839	0.858	0.848	0.854	0.797	0.803	0.800	0.816

Table 4. Effects of the clustering parameters  $\alpha$  and  $\theta$  on polarity classification.

To verify the theoretical parameter optimization, we conducted an experiment to examine the effects of  $\alpha$  and  $\theta$  on polarity classification. The results are listed in Table 4.

As can be seen from Table 4, the experimental results conform to the theoretical optimization. The F-score reach the largest for mobilephone domain when  $\theta=0.15$  and  $\alpha=0.8$ , while the corresponding real best values of  $\theta$  and  $\alpha$  are 0.3 and 0.6 for the car domain. Furthermore, it is also observed that larger value of  $\theta$  and smaller value of  $\alpha$  is beneficial to polarity classification for mobilephone domain while the trend is reversed for car domain. The reason may be that mobilephone products have less attributes than car products, suggesting a looser clustering standard for mobilephone domain. Moreover, looser standard will result in less number of clusters after review clustering, and in this case literal similarity will contribute more to review clustering. That is why mobilephone review clustering has a larger interpolation coefficient than car review clustering.

In addition to the above two clustering parameters, we have also conducted an experiment to examine the effect of the number of generated paraphrases on polarity classification. The results are shown in Figure 3.

Figure 3 reveals that the influence of paraphrase generation on polarity classification is changing with the number of generated paraphrases. When the number of generated

paraphrases is less than 10, the F-score for polarity classification fluctuates with the number of generated paraphrases. However, when the number exceeds 100, the F-score will consistently rise with the number of generated paraphrases. The reason might be due to the fact that the noise introduced by paraphrase generation may have a relatively greater negative impact on polarity classification in case of the small size of paraphrase generation.

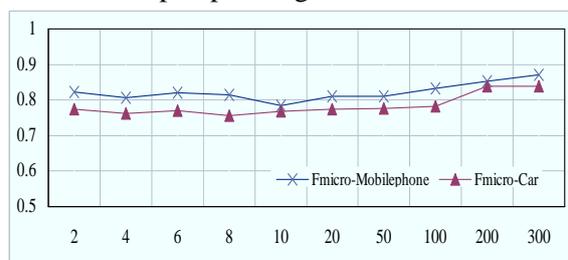


Figure 3. Effects of the number of generated paraphrases on polarity classification.

### 4.3 Experimental Results

In order to evaluate the effectiveness of the cluster-based method with multi-classifiers from the expanded review clusters (viz. M\_SVM+Para), our experiment also involves three baselines for comparison, namely the traditional separate SVM classifier from the original training corpora in Table 1 (viz. S-SVM) or from the expanded original corpora via paraphrase generation (viz. S\_SVM+Para), the cluster-based method with multiple SVM classifiers built from the review clusters without

paraphrasing (viz. M-SVM). The experimental results are listed in Table 5 and Table 6.

Methods	P <sub>macro</sub>	R <sub>macro</sub>	F <sub>macro</sub>	F <sub>micro</sub>
S_SVM	0.831	0.855	0.843	0.840
M_SVM	0.815	0.828	0.822	0.832
S_SVMs + Para	0.847	0.870	0.858	0.859
M_SVMs + Para	<b>0.856</b>	<b>0.874</b>	<b>0.865</b>	<b>0.871</b>

Table 5. Results for the mobilephone domain data.

Methods	P <sub>macro</sub>	R <sub>macro</sub>	F <sub>macro</sub>	F <sub>micro</sub>
S_SVM	0.775	0.764	0.769	0.781
M_SVM	0.760	0.748	0.754	0.779
S_SVMs+Para	0.788	0.791	0.789	0.804
M_SVMs + Para	<b>0.827</b>	<b>0.813</b>	<b>0.820</b>	<b>0.838</b>

Table 6. Results for the car domain data.

From these results, we have several observations. First, the cluster-based system with paraphrasing yields the best performance for both domains, illustrating the benefits of opinion clustering and paraphrasing to polarity classification. Second, we can observe that the performance degrades when applying the clustering-based method to polarity classification without paraphrase generation. The reason may be due to the fact that the training data become too small for some clusters after review clustering. Finally, using opinion paraphrase generation results in consistent increasing of performance for the two datasets in use, showing in a sense opinion paraphrasing facilitates a effective way to expand training corpora for sentiment analysis.

## 5 Conclusions and Future Work

In this paper we present a new opinion cluster based framework that uses multiple cluster-based SVM classifiers to perform polarity classification of short product reviews. The main contributions of this paper are: (1) the idea of jointly using opinion clusters and paraphrases to explore richer contextual information or specific cues in short text for sentiment analysis; (2) the demonstration that opinion clustering and paraphrasing are of great value to polarity classification of short text like online product reviews.

For future work, we intend to exploit a more tailored method to achieve high-quality opinion clustering and paraphrase generation for polarity classification. Furthermore, we also plan to extend our current method to other feature

representations like the emerging distributed vector representations or apply our system to other languages like English.

## Acknowledgments

This study was supported by National Natural Science Foundation of China under Grant No. 61170148 and the Returned Scholar Foundation of Heilongjiang Province.

## References

- Anthony Fader, Luke Zettlemoyer, and Oren Etzioni. 2013. Paraphrase-driven learning for open question answering. In *Proceedings of ACL'13*, pages 1608-1618.
- Bo Pang, and Lillian Lee. 2008. Opinion mining and sentiment analysis. *Foundations and Trends in Information Retrieval*, 2(1-2): 1-135.
- Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan. 2002. Thumbs up?: Sentiment classification using machine learning techniques. In *Proceedings of EMNLP'02*, pages 79-86.
- Chih-Chung Chang, and Chih-Jen Lin. 2011. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2(27): 1-27.
- Cícero Nogueira dos Santos, and Maíra Gatti. 2014. Deep convolutional neural networks for sentiment analysis of short texts. In *Proceedings of COLING'14*, pages 69-78.
- Guohong Fu, and Xin Wang. 2010. Chinese sentence-level sentiment classification based on fuzzy sets. In *Proceedings of COLING'10*, pages 312-319.
- Guohong Fu, Chunyu Kit, and Jonathan J. Webster. 2008. Chinese word segmentation as morpheme-based lexical chunking. *Information Sciences*, 178(9): 2282-2296.
- Guohong Fu, Yu He, Jiaying Song, and Chaoyue Wang. 2014. Improving Chinese polarity classification via opinion paraphrasing. In *Proceedings of CLP'14*, pages 35-42.
- Hyun Duk Kim, Malú G. Castellanos, Meichun Hsu, ChengXiang Zhai, Umeshwar Dayal, and Riddhiman Ghosh. 2013. Ranking explanatory sentences for opinion summarization. In *Proceedings of SIGIR'13*, pages 1069-1072.
- James Bergstra, and Yoshua Bengio. 2012. Random search for hyper-parameter optimization. *The Journal of Machine Research*, 13(1): 281-305.
- Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. Glove: Global vectors for word

- representation. In *Proceedings of EMNLP'14*, pages 1532-1543.
- Kang Liu, Liheng Xu, and Jun Zhao. 2014. Extracting opinion targets and opinion words from online reviews with graph co-ranking. In *Proceedings of ACL'14*, pages 314-324.
- Michael Heilman, Noah A. Smith. 2010. Tree edit models for recognizing textual entailments, paraphrases, and answers to questions. In *Proceedings of NAACL'10*, pages 1011-1019.
- Michael Speriosu, Nikita Sudan, Sid Upadhyay, and Jason Baldridge. 2011. Twitter polarity classification with label propagation over lexical links and the follower graph. In *Proceedings of the First workshop on Unsupervised Learning in NLP*, pages 53-63.
- Mining Hu, and Bing Liu. 2004. Mining and summarizing customer reviews. In *Proceedings of SIGKDD'04*, pages 168-177.
- Nitin Madnani, and Bonnie J. Dorr. 2010. Generating phrasal and sentential paraphrases: A survey of data-driven methods. *Computational Linguistics*, 36(3): 342-387.
- Peter D. Turney. 2002. Thumbs up or thumbs down?: Semantic orientation applied to unsupervised classification of reviews. In *Proceedings of ACL'02*, pages 417-424.
- Rada Mihalcea, Carmen Banea, Janyce Wiebe. 2007. Learning multilingual subjective language via cross-lingual projections. In *Proceedings of ACL'07*, pages 976-983.
- Rahul Bhagat, and Eduard Hovy. 2013. What is a paraphrase? *Computational Linguistics*, 39(3): 463-472.
- Rishabh Mehrotra, Rushabh Agrawal, and Syed Aqueel Haider. 2012. Dictionary based sparse representation for domain adaptation. In *Proceedings of CIKM'12*, pages 2395-2398.
- Shiqi Zhao, Xiang Lan, Ting Liu, and Sheng Li. 2009. Application-driven statistical paraphrase generation. In *Proceedings of the ACL-IJCNL'09*, pages 834-842.
- Sida Wang and Christopher D. Manning. 2012. Baselines and bigrams: Simple, good sentiment and topic classification. In *Proceedings of ACL'12*, pages 90-94.
- Svitlana Volkova, Theresa Wilson, David Yarowsky. 2013. Exploring sentiment in social media: Bootstrapping subjectivity clues from multilingual twitter streams. In *Proceedings of ACL'13*, pages 505-510.
- Tetsuji Nakagawa, Kentaro Inui, and Sadao Kurohashi. 2010. Dependency tree-based sentiment classification using CRFs with hidden variables. In *Proceedings of HLT-NAACL'10*, pages 786-794.
- Theresa Wilson, Janyce Wiebe, and Paul Hoffmann. 2009. Recognizing contextual polarity: An exploration of features for phrase-level sentiment analysis. *Computational Linguistics*, 35(3):99-433
- Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781.
- Yu He, Da Pan, and Guohong Fu. 2015. Chinese explanatory segment recognition as sequence labeling. *Communications in Computer and Information Science*, 503: 159-168.

## Automatic Classification of Spoken Languages using Diverse Acoustic Features

**Yaakov HaCohen-Kerner**

Dept. of Computer Science  
Jerusalem College of Technology –  
Lev Academic Center  
21 Havaad Haleumi St., P.O.B. 16031  
9116001 Jerusalem, Israel  
kerner@jct.ac.il

**Ruben Hagege**

Dept. of Electronics  
Jerusalem College of Technology –  
Lev Academic Center  
21 Havaad Haleumi St., P.O.B. 16031  
9116001 Jerusalem, Israel  
hagege.ruben@gmail.com

### Abstract

Many of the language identification (LID) systems are based on language models using machine learning (ML) techniques that take into account the fluctuation of speech over time, such as Hidden Markov Models (HMM). Considering the fluctuation of speech results LID systems use relatively long recording intervals to obtain reasonable accuracy. This research tries to extract enough features from short recording intervals in order to enable successful classification of the tested spoken languages. The classification process is based on frames of 20 milliseconds (ms) where most of the previous LID systems were based on much longer time frames (from 3 seconds to 2 minutes). We defined and implemented 173 low level features divided into three feature sets: cepstrum, relative spectral (RASTA), and spectrum. The examined corpus, containing speech files in seven languages, is a subset of the Oregon Graduate Institute (OGI) telephone speech corpus. Six machine learning (ML) methods have been applied and compared and the best optimized results have been achieved by Random Forest (RF): 89%, 82%, and 80% for 2, 5, and 7 languages, respectively.

### 1 Introduction

LID is used either as a standalone task or as a pre-processing step, capturing the first seconds (sec) of the recording and processing it in order to transfer

the control to the appropriate next stage; e.g. speech recognition systems, multilingual translation systems or call-centers (e.g., emergency calls) routing, where the response time of a native operator might be critical.

LID is a process by which a given spoken utterance language is automatically identified (Muthusamy et al., 1994). Most LID systems are based on high level features such as frequency of a single phoneme, phoneme sequences (Zissman and Singer, 1994), syllable, words, and prosody (Thymé-Gobbel and Hutchins, 1996). Such LID systems need a comprehensive corpus, including transcription from trained humans, and long enough intervals to correctly classify, first, these high level features and then the spoken language (Zissman, 1996; Greenberg, 1999). Any error in the higher level feature recognizers is carried over, and probably/possibly amplified in, the following steps. However, providing a comprehensive corpus enables higher level features which ensure better results than using acoustic features alone. LID systems based on higher level features have one principal problem: Tokenizing those features accurately has proven to be the main obstacle thus far in high accuracy of natural LID (Abramson, 2003). Matejka et al. (2005) found that separating gender before processing improved the LID's accuracy.

A LID system has two main parts: feature extraction, where a vector of measurements that

should characterize the high level features are extracted from the signal; and pattern matching, where these extracted features are processed using statistical (like in this study) or temporal (Rabiner, 1989) methods to recognize speech languages. The approach taken in our study does not resort to the use of phoneme recognizers or any higher level features. Instead, we rely on low-level features alone, rather than using low-level features to predict intermediate features as in previous work. The motivation is "quicker response time and simpler training stages".

The rest of this paper is organized as follows: Section 2 presents an overview of previous LID systems. Section 3 describes the different feature sets chosen for this study. Section 4 presents the suggested classification model and the implemented features for LID of seven languages: French (FR), Farsi (FA), Japanese (JA), Korean (KO), Mandarin (MA), Tamil (TA), and Vietnamese (VI). Section 5 describes the examined corpora and experimental results and analyzes them. Section 6 includes a summary and proposes suggestions for future research.

## 2 Previous LID Systems

In this section, we focus our overview of previous LID systems that had goals similar to our work or systems that used the same (or a very similar) corpus and / or set of languages.

Silences are an integral part of speech recordings in all languages. These silences are usually unnecessary for computer processing purposes: they considerably increase the files size and potentially lead to a great loss of accuracy of the LID system. Thus, the first step in most LID systems use a Voice Activation Detection (VAD), a sub-process that identifies and discards those silences. Other factors must also be taken in account, such as the channels through which the speech is conveyed. These channels add noises to the speech which, although it is still recognizable by Humans, causes difficulties for computers. Therefore, to ensure better performance using ML methods, a noise-filtering sub-process is preferable. All the previous LID systems described below used at least one of those techniques to enhance their results. Thus, we decided to implement those techniques as well.

Hazen and Zue (1993) tested their system on the OGI Multi-Language telephone speech (MLTS)

corpus (Yeshwant K. Muthusamy et al., 1992). Using both genders on the speech utterances. The average length of selected utterance on the OGI corpus is about 13.4 sec. They developed and tested a LID system based on a segment-based approach composed of phonotactic (Matejka et al., 2005), prosodic, and acoustic property of the languages. The features used are 14 Mel Frequency Cepstral Coefficients (MFCC), in contrast to most LID systems that use 13 MFCCs, for each frame. The Cepstral Coefficient (CC) deltas were also extracted along with the pitch (F0) feature, which was used to find and discard silences (VAD) as well as removing the speaker dependency. Each frame was 5ms long. They tested their system on 10 languages, an overall system performance of 48.6% was achieved using n-grams, acoustic, duration, F0, and delta-F0 features. The correct language was one of the top three choices 74.4% of the time. Their results on less than a sec for each file is between 10% and 20%.

Muthusamy et al. (1993) based their system on the OGI-MLTS corpus with 13.4 sec of speech per file on average. They explained that at the time it was still not clear which of the possible LID techniques will be more suitable to discriminate languages. Thus, they compared 3 different approaches (acoustic features, category segmentation, and phonetic classification). In all the sets, the Perceptual Linear Predictive (PLP; Dave, 2013) coefficients was applied using 10ms frames with either 4ms or 7ms of overlapping intervals. Their best result was obtained using 200 bigrams and unigrams. They classified the whole speech files (up to 50 sec) using these feature sets and the Artificial Neural Network (ANN; Lopez-Moreno et al., 2014) ML method. Best results of 86.3% on 2 languages (EN and JA) were obtained. They also obtained 70% accuracy using acoustic features (PLP) alone.

Lamel and Gauvain (1994) presented a LID system tested on the OGI corpus and Laboratory quality speech (four different corpora, two for EN and two for FR language). They applied phone-based acoustic likelihoods, using parallel-trained Hidden Markov Models (HMMs). In 10 languages classification tasks, they tested the OGI corpus and got 48.7%, 55.1%, and 59.7% on intervals of 2, 6 and 10 sec, respectively. On 2 languages (FR and EN) however, their results rose to 76%, 80.87%, and 81.33% on 2, 6, and 10 sec, respectively.

Shuichi and Liang (1995) tested their system on corpora produced from multiple respected sources,

containing the OGI, NTT and NATC corpora. They proposed a LID system based solely on F0 and its time-dependent patterns using discriminant analysis on the polygonal line approximation of the F0 patterns. Using the 21 extracted features from the F0 behavior (e.g., slope, shape, etc.) They achieved 75% on the NTT and NATC corpus and 63.3% on the OGI corpus.

Zissman (1996) compared different LID techniques on the OGI corpus. he also uses RelAtive SpecTrAl (RASTA; Hermansky and Morgan, 1994) as a part of the pre-processing of speech in order to remove slowly varying, linear channel effects from the raw feature vectors. He obtained that single-language phone recognition followed by language-dependent language modeling (PRLM) gave best results when distinguishing 10 languages, giving results as high as 79% on 45 sec speech utterances and 63% on 10 sec. Furthermore, their results in 2 languages discrimination were up 97% on 45 sec of speech (EN and SP) using parallel phone recognition (PPR; Nagarajan and Murthy, 2004) and 90% on 10 sec (JA and SP) using parallel PRLM, they also tested Gaussian Mixture Model (GMM) achieving 84% on 10 sec long audio file (EN and JA).

Lippmann (1997) compared human and state of the art LID available at the time and noted that even if machine ability to identify a language was still several order of magnitude lower than human, he only proved that it was needed to work on more reliable, noise robust, LID systems and components. "The transcription error rate (ER) is less than 0.009% for read digits, less than 0.4% for read sentences from the Wall Street Journal, and less than 4% for spontaneous conversations recorded over the telephone." His study was focused more on isolated digits or alphabet letters recognition in order to perform LID than spontaneous conversation.

Pellegrino and Andre-Obrecht (2000) tested a LID system on 5 languages from the OGI-MLTS corpus: FR, KO, VI, JA, and SP. Using two different approach (GMM and HMM) to model either the vocalic (GMM) or phonetic (HMM) space. Features such as MFCC (8 coefficients) and duration of the segments obtained using a so called "Forward Backward Divergence" (Andre-Obrecht, 1988) segmentation algorithm. The features are extracted inside segments by frames of 20ms. The purpose of this study was to demonstrate the possibility to extract vowel information from acoustic signal.

Results were presented either in segments of 2 minutes or 45 sec of speech. Their best results are 73.8% and 61.2% on 4 and 5 languages, respectively, using 2-minute-long speech utterances and all of the features presented earlier.

Kirchhoff and Parandekar (2001) based her LID system on the OGI corpus. Using Multi-Stream Statistical N-Gram Modeling, he compared the accuracy of the model on different speech lengths (from 3 to 45 sec). Features such as manner, consonantal place, vowel place, front-back, and rounding and their dependencies (front-back -vowel place and front-back – consonantal place) were used. On 10 languages, her results were as high as 48%, 58.8%, and 64.6% on audio files of less than 3 sec, between 3 and 15 sec, and longer than 15 sec audio files respectively.

Torres-carrasquillo et al. (2002) used the 1996 Linguistic Data Consortium's CallFriend LID evaluation set, a 12 languages corpus that was allocated as follows: The development set consists of 1184 30-sec utterances and the evaluation set of the corpus consists of 1492 30-sec utterances, each distributed among the various languages of interest. LID was performed using GMM Tokenization: extracting features to then tokenize them using GMM and finally perform LM (in an attempt to enhance the PRLM system developed by Zissman in 1996). Using the evaluation set, an ER of 17% (83% of accuracy) was obtained using both Parallel-PRLM, GMM tokenizers, and GMM acoustics.

Li et al. (2007) investigate automatic spoken language identification (LID) process based on Vector Space Modeling (VSM; e.g., Martínez et al., 2011). The evaluation is carried out on recorded telephone speech of 12 languages: Arabic, EN, FA, FR, GE, Hindi (HI), JA, KO, MA, SP, TA, and VI from 1996 and 2003 NIST Language Recognition Evaluation. Achieving ER as low as 2.75% (97.25% of accuracy) on 30-sec of speech on 6 languages identification. The 2<sup>nd</sup> focus in their project was the possibility of Real-time (RT) applications.

All those studies based their performance evaluation on a wider time frame than ours, this is a major difference, and it must be considered when comparing our results. Moreover, unlike most of the previous works, our system is not designed to classify languages using keyword, phoneme, or even vowel recognition. It doesn't require any language model either, making the language training process a lot faster.

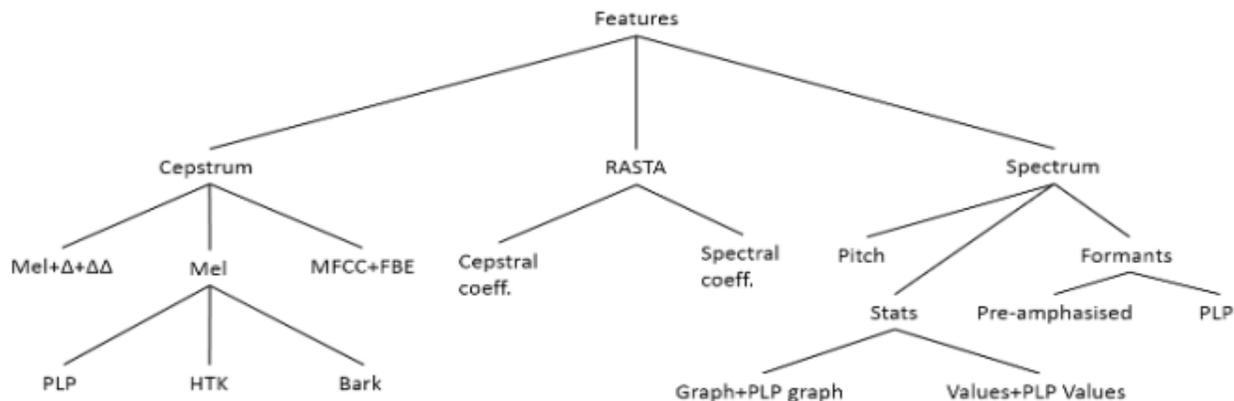


Figure 1. The computed acoustic features.

### 3 Acoustic Features

In this research, we consider 173 acoustic features divided into three main feature sets: 114 Cepstrum features, 28 RASTA features, and 30 Spectrum features. The hierarchical structure of the three feature sets is described in Figure 1. Although most of these features have been extensively used in previous LID systems, these features were a basis for higher level features. In contrast, our system is solely relying on an extensive combination of low level features which has never been used before to the best of our knowledge.

The Cepstrum features set is composed of groups of coefficients which represent the filter sources (e.g., shape of the mouth etc.). The Bark and Mel scales (Stevens et al., 1937; Stevens and Volkman, 1940) are perceptual scales of the pitch. Filter Bank Energy (FBE) represents the energy from all the band filters (Huang et al., 2001) used to extract the MFCCs. HTK (HMM ToolKit) represents the CCs extracted using parameters close to the original HTK (Young et al., 2002; Ellis, 2005; Brookes, 1997) approach.

The RASTA set represents features extracted after filtering. These features are extracted in both spectrum and cepstrum, taking cepstrum coefficients using both Linear Predictive Coefficients (LPC), which are used to compute spectral and cepstral features, and RASTA filter.

We implemented the IIR RASTA filter as it is described in Equation 1 (Ellis 2005; Matlab RASTA's filter transfer function implementation).

$$H(z) = 0.1 \times \frac{2z^5 + z^4 - z^2 - 2z}{z - 0.94} \quad (1)$$

The -0.94 weight in the denominator side was chosen in our Matlab implementation to improve filter response time from the original 500ms to 160ms response time using -0.98 that is applied in some of the previous works (Zissman, 1996).

The Spectrum features set consists of the following feature sets: (1) The pitch (F0) feature (Titze, 1994; Zahorian and Hu, 2008). (2) The graph features, which are statistical features that record the occurrence of each frame's median peak. (3) Values (mean, median, min, max, std), and frequency (median) stats, describing each frame's FFT. (4) Formants are the principal spectral component of a frame, defined by "the spectral peaks of the voice spectrum". Linguists largely maintain that the first two formants (in EN at least) are sufficient to differentiate between all vowels (Ladefoged and Johnson, 2014). We decided to extract the 4 first formants.

There is a spectral tilt in speech caused by the voice-source (vocal tract). The vocal tract creates the formant frequencies, so when these are estimated (using FFT), the spectral tilt needs to be removed. This is usually done with a simple pre-emphasis filter, as in our case.

The algorithms that were developed, using MATLAB (V8.3), for this study were built for feature extraction, VAD, and WEKA interfacing purposes. They were designed to perform for real-time applications and, in addition, to be dynamic so that they could be easily changed to extract any specific set of features and/or classes. WEKA (Hall et al., 2009) explorer was used for the classification task.

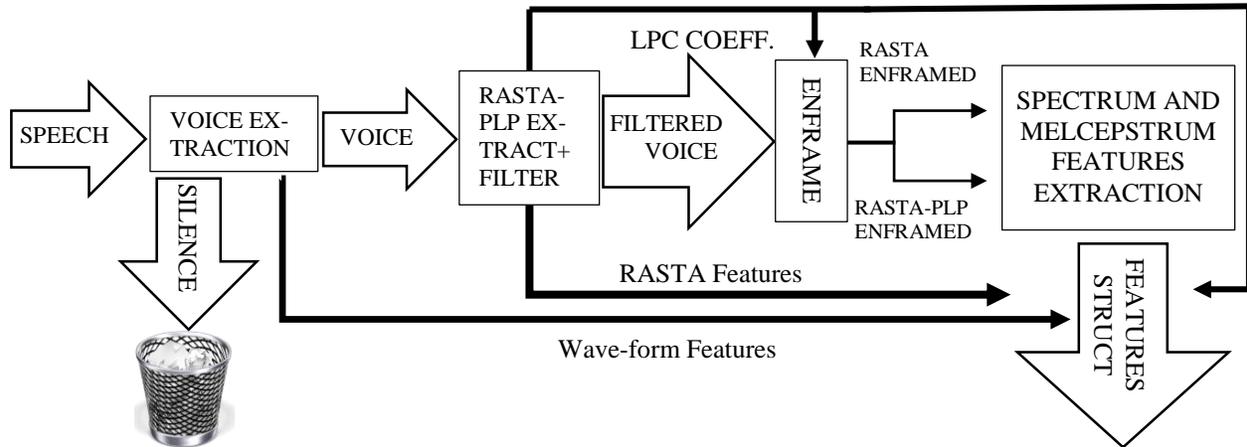


Figure 2. The feature extraction process (stages 2-3 in the classification model).

#### 4 The Classification Model

The main stages of the classification model are as follows:

1. Building the speech corpus (Table 1).
2. Cleaning the speech files. Removing the silent intervals and filtering each file (Figure 2).
3. Computing the features for each file (Figure 2).
4. Transforming the features matrix into a WEKA input file.
5. Applying six ML methods on various combinations of feature sets using WEKA.

Figure 2 describes the feature extraction process (stages 2-3 in the classification model). This Figure grossly illustrates how the structure containing the features, used to discriminate the languages, is extracted. In order to process the speech files as clean as possible equalization and filtering seemed appropriate to better distinguish noise or silence from speech (experimentation shows an improvement of at least 5% in VAD classification after RASTA filtering compared to before).

A RASTA filter is applied to suppress the effect of the telephone line on the features. First, the audio file (speech) is passed through a VAD, and the silence intervals are discarded. One of the features used to perform the VAD (F0) is also extracted (Zahorian and Hu, 2008a). Speech, rid of silences, goes through RASTA feature extraction that extracts the RASTA features family and filters the audio files. The filtered, silence-free speech file is then enframed (Brookes and others, 1997) into frames of 20ms with 10ms overlap, and a Hamming window is applied on each frame (where the last frame is discarded if shorter than 20ms). The frames are sent to the spectrum and cepstrum features extraction

where remaining features are extracted. Then, the features extracted are grouped together inside a “features structure” with each frame’s features contained in a single line vector. Every file, after completing the feature extraction process, outputs a structure composed of X vectors (depending on file length) containing the 173 features. The resulting structure is then converted into a matrix, and the matrices are concatenated so that every language gets a part of all the files (presented experimented on gets 10,000 feature vectors (frames) for each language).

Six supervised ML methods including one decision tree, two ensemble learning, and two SVMs, have been selected for application of the last stage in our model:

1. J48 is an improved variant of the C4.5 decision tree induction (Quinlan, 1993; Quinlan, 2014) implemented in WEKA. J48 is a classifier that generates pruned or unpruned C4.5 decision trees. The algorithm uses greedy techniques and is a variant of ID3, which determines at each step the most predictive attribute, and splits a node based on this attribute. J48 attempts to account for noise and missing data. It also deals with numeric attributes by determining where thresholds for decision splits should be placed. The main parameters that can be set for this algorithm are the confidence threshold, the minimum number of instances per leaf and the number of folds for REP. As described earlier, trees are one of the easiest thing that could be understood because of their nature.
2. RF, an ensemble learning method for classification and regression (Breiman, 2001). This ML technique is an ensemble learning

technique. Ensemble methods use multiple learning algorithms to obtain better predictive performance than what could be obtained from any of the constituent learning algorithms. RF is based on what's called a random tree: a tree that randomly chooses  $K$  attributes and then build a simple tree with no pruning. RF let us choose the number of features ( $K$ ) and the number of random trees ( $I$ ) we want to use.

3. MultiBoostab (MB) (Webb, 2000) is an extension to the highly successful AdaBoost (Freund and Schapire, 1996) technique for forming decision committees. MB technique can be viewed as combining AdaBoost with wagging (Bauer and Kohavi, 1999). It is able to harness both AdaBoost's high bias and variance reduction with wagging's superior variance reduction. Using C4.5 as the base learning algorithm, Multi-boosting is demonstrated to produce decision committees with lower error than either AdaBoost or wagging significantly more often than the reverse. It offers the further advantage over AdaBoost of suiting parallel execution. In WEKA, the default base classifier for MB is Decision Stump (Iba and Langley, 1992).
4. BayesNet (BN) is a variant of a probabilistic statistical classification model that represents a set of random variables and their conditional dependencies via a directed acyclic graph (DAG) (Friedman et al., 2000; Heckerman, 1997; Pourret, 2008).
5. Logistic regression (LR) (Cessie et al., 1992) is a variant of a probabilistic statistical classification model that is used for predicting the outcome of a categorical dependent variable (i.e., a class label) based on one feature or more (Landwehr et al., 2005; Sumner et al., 2005).
6. Sequential Minimal Optimization (SMO; Platt, 1998; Keerthi et al., 2001) is a variant of the Support Vectors Machines (SVM) ML method (Cortes and Vapnik, 1995). The SMO technique

is an iterative algorithm created to solve the optimization problem often seen in SVM techniques. SMO divides this problem into a series of smallest possible sub-problems, which are then resolved analytically.

These ML methods have been applied using the WEKA platform (Frank, 2006; Hall et al., 2009). We performed parameter tuning with Info-Gain (IG), a feature selection metric for classification purposes. Yang and Pedersen (1997) reported that IG performed best in their multi-class benchmarks. The accuracy of each model was estimated by a 10-fold cross-validation test.

## 5 Experimental Results

The OGI Multi-language Telephone Speech Corpus (Muthusamy et al., 1992; Muthusamy et al., 1993) consists of telephone speech recorded in eleven languages: EN, FA, FR, GE, HI, JA, KO, MA, SP, TA and VI. The OGI corpus is not balanced between males and females: the male files represent more than 75% of the corpus. Thus, in this study, we only used the male speech files. The examined corpus contains 478 files (each from a different person) from seven selected languages with an average length of 44.3 sec, each file consists of free, continuous speech.

Since our classification system was heavily consuming a classic workstation's RAM, the final corpus had to be reduced to 10,000 frames per language (equally distributed on the various files), that are equivalent to 100 sec of speech. As most of telephone speech corpus based LID systems (Hermansky, 2011), we used a RASTA filter (Matlab implementation; Ellis, 2005) to reduce the channel (telephone) effect noises.

Table 1 presents general information about the speech files contained in the examined corpus. The number of speech files for each language is ranging from 53 to 86. The average time length is rather similar for all languages (from 42.2 to 47.5 sec).

#	Language	# of speech files	Length of speech files in sec.	Avg. time length in sec.
1	French (FR)	55	37<x<49	47.5
2	Farsi (FA)	81	5<x<49	44.4
3	Japanese (JA)	53	23<x<49	46.6
4	Korean (KO)	62	4<x<49	42.2
5	Mandarin (MA)	73	10<x<49	42.5
6	Tamil (TA)	86	8<x<49	44.3
7	Vietnamese (VI)	68	7<x<49	43.9

Table 1. General information about the speech files selected from the OGI corpus.

#	Languages	BN	SMO	LR	MB	J48	RF
2	FR, TA	66.47	72.59	73.02	66.84	80.21	<b>88.27</b>
3	FR, MA, TA	54.25	58.76	60.41	42.96	68.47	<b>81.17</b>
4	FR, MA, TA, VI	45.99	50.00	51.04	34.11	62.72	<b>77.51</b>
5	FR, FA, MA, TA, VI	36.84	42.81	43.34	27.45	57.03	<b>73.97</b>
6	FR, FA, JA, MA, TA, VI	32.36	37.54	37.70	22.89	53.29	<b>71.83</b>
7	FR, FA, JA, KO, MA, TA, VI	29.38	33.52	33.66	19.48	51.50	<b>71.13</b>

Table 2. Accuracy results for the best language combinations using default parameters and all features.

For each tested combination of feature sets we applied all of the 6 chosen ML methods: BN, SMO, LR, MB, J48 and RF. We then checked our feature sets using IG, among other feature selection methods, and no features with zero weights were found. We also performed a parameter tuning process in order to achieve the best results on the best default ML method (see Figure 3). All the optimized results are obtained as follows: each ML parameter is tuned in a hill climbing fashion, changing one parameter at a time (manually) until the best value is obtained (within a <1% margin). On ML methods based on simple trees such as J48, it appears to be enough: the parameters seemed to be independent (according to the results we had). However, for the RF ML method, the two principal parameters were tuned together since our preliminary results tends to show that they have an influence on one another.

Unlike previously developed methods (see Section 2) that focus on changes of specific features over time to classify languages, our research assess the potential of features computed on a single frame (20ms), using each frames as a basis of the classification decision.

Table 2 presents the accuracy results for the 6 selected ML methods under default parameters proposed by the WEKA platform. The best language combinations from 7 to 2 languages (with accuracy as the deciding factor) were selected by analyzing the confusion matrices that were produced by the best ML method – RF (according to Table 2), and filtering out the less successful language in each stage. Firstly, The RF ML method has been applied on the all seven languages and then the six best languages (achieving the best accuracy) were picked from those seven based on the confusion matrix, and so on, until only the best combination of two languages remains. As a result, we got the following language combinations:

7. FR, FA, JA, KO, MA, TA, and VI.
6. FR, FA, JA, MA, TA, and VI.
5. FR, FA, JA, TA, and VI.
4. FR, JA, TA, and VI.
3. FR, JA, and TA.
2. FR, and TA.

Various conclusions concerning our LID system can be drawn from Table 2: (1) The RF method obtained the best accuracy results. (2) The 2<sup>nd</sup> best ML method was J48. (3) The decision tree ML methods are the best ML methods for our LID tasks.

Since RF is uncontestably the most suited technique between the six chosen ML techniques, we decided to optimize the RF’s parameters (maxDepth, numFeatures, numTrees, and seed). Because of the lack of space to display results, we were only able to present optimized results on a limited set of languages. We chose to optimize the best language combinations of size 2, 5, and 7 (see Table 2). All the optimized results are obtained as follows: each parameter is tuned in a hill climbing fashion. By manually changing one parameter at a time till the best value is obtained within a reasonable (<0.1%) margin.

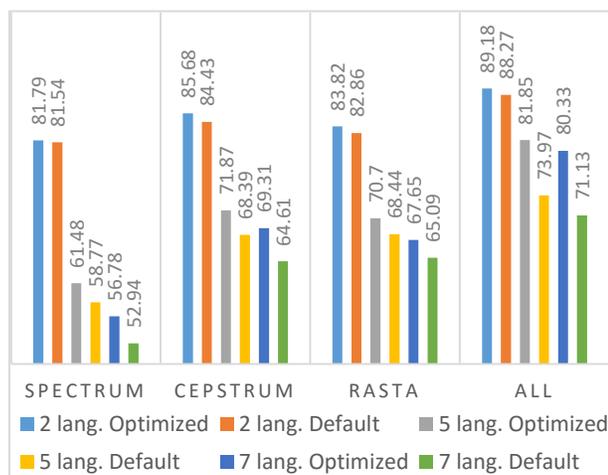


Figure 3. Optimized/default accuracy on each feature set and all features.

Multiple conclusions can be drawn from Figure 3: (1) RF has a great optimizing potential, (2) The more language it classifies, the greater become the optimization over default results, (3) The Cepstrum feature set has the greatest differentiation potential. A possible explanation for these results can be the high number of relevant features: the more relevant data one have, the easier classification become. (4) RASTA has the greatest differentiation potential per feature; its performance is almost equal to the Cepstrum set while using only a quarter of its number of features.

## 6 Summary and Future Research

In this paper, we present a methodology for classifying speech files from 7 different languages based on combined cepstrum, RASTA, and spectrum feature sets. This methodology compares six different ML methods. RF, the best ML method achieves relatively high accuracy results of 89.18%, 81.85%, and 80.33% for the following classification experiments: 2, 5, and 7 best language combinations, respectively.

The novelties of this research are in its reliance: (1) on low-level features alone, rather than using low-level features changes over time to predict intermediate features as in previous work, and (2) on much smaller frames (20ms) in comparison to most previous LIDs whose results are based on much longer time periods (at least 3 sec. or longer; see Martinez et al., 2013, among many other references below, for detail on the impact of frame length on result). Eliminating reliance on intermediate features is an important contribution, especially for low-resource languages.

Our results are comparable to the accuracy level of top LID systems from about 20 years ago (that also used different versions of the OGI corpus; see section 2). However, our LID system uses a time frame that is at least 60 times shorter than the time frames used by previous LID systems. To the best of our knowledge, there is no LID system which is based on a such short time frame.

Future directions for research are: (1) Developing additional feature sets in general and additional features in particular (with an emphasis on the RASTA set), (2) Applying other ML methods in order to find the most suited method for LID purposes, (3) Conducting more experiments using more speech files from more languages, (4)

Discovering which combination of features in particular are appropriate for LID of speech files using the system we developed, and (5) How well does the system based on acoustic features work for non-native speakers?

## Acknowledgments

The authors would like to thank Shmuel Kirshner for his many advises on theory of speech processing, Edmond Shalom for enabling us to start this research, Shlomo Engelberg for his continuous support, on each aspect of this endeavor, Shimon Mizrahi for giving us the time needed to accomplish such a work, Boris Dekhovitch for his comments, Evgeni Frishman and Yaakov Friedman for financing of the database. Many thanks to the Dept. of Electronics and the rector Kenneth Hochberg of the Jerusalem College of Technology, Lev Academic Center, for their assistance during this research. We would also like to thank the three reviewers for their useful and instructive comments.

## References

- Arthur S. Abramson. 2003. *A Practical Introduction to Phonetics (review)*. volume 79. Clarendon Press Oxford.
- Régine Andre-Obrecht. 1988. A New Statistical Approach For The Automatic Segmentation Of Continuous Speech Signals. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 36(1):29–40, January.
- Eric Bauer and Ron Kohavi. 1999. An Empirical Comparison of Voting Classification Algorithms: Bagging, Boosting, and Variants. *Machine Learning*, 36(1):105–139.
- Leo Breiman. 2001. Random Forests. *Machine Learning*, 45(1):5–32.
- Mike Brookes. 1997. Voicebox: Speech Processing Toolbox for Matlab. ... *From Www. Ee. Ic. Ac. Uk/Hp/Staff/Dmb/Voicebox/Voicebox* ...
- Saskia Le Cessie, J. C. Van Houwelingen, and Royal Statistical Society. 1992. Ridge Estimators in Logistic Regression. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 41(1):191–201.

- Corinna Cortes and Vladimir Vapnik. 1995. Support-Vector Networks. *Machine learning*, 20(3):273–297.
- Namrata Dave. 2013. Feature Extraction Methods LPC, PLP and MFCC In Speech Recognition. *International Journal for Advance Research in Engineering and Technology*, 1(Vi):1–5.
- Daniel P.W. Ellis. 2005. PLP and RASTA (and MFCC, and Inversion) in Matlab. [Http://Www.Ee.Columbia.Edu/~Dpwe/Resources/Matlab/Rastamat/](http://Www.Ee.Columbia.Edu/~Dpwe/Resources/Matlab/Rastamat/).
- Joachim Frank. 2006. *Electron Tomography: Methods for Three-Dimensional Visualization of Structures in the Cell*. Morgan Kaufmann.
- Yoav Freund and Re Robert E Schapire. 1996. Experiments with a New Boosting Algorithm. In *International Conference on Machine Learning*, volume 96, pages 148–156.
- Nir Friedman, M Linial, I Nachman, and D Pe’er. 2000. Using Bayesian Networks to Analyze Expression Data. *Journal of computational biology : a journal of computational molecular cell biology*, 7(3-4):601–620.
- Steven Greenberg. 1999. Speaking in Shorthand - a Syllable-Centric Perspective for Understanding Pronunciation Variation. *Speech Communication*, 29(2):159–176.
- Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, and Ian H. Witten. 2009. The WEKA Data Mining Software. *ACM SIGKDD Explorations Newsletter*, 11(1):10.
- Timothy J Hazen and Victor Zue. 1993. Automatic Language Identification Using a Segment-Based Approach. In *3rd International Conference on Spoken Language Processing*, pages 1307–1310.
- David Heckerman. 1997. Bayesian Networks for Data Mining. *Data Mining and Knowledge Discovery*, 119(1):79–119.
- Hynek Hermansky. 2011. Speech Recognition from Spectral Dynamics. *Sadhana - Academy Proceedings in Engineering Sciences*, 36(5):729–744.
- Hynek Hermansky and Nelson Morgan. 1994. RASTA Processing of Speech. *Speech and Audio Processing, IEEE Transactions on*, 2(4):578–589.
- and Raj Foreword By-Reddy. Huang, Xuedong, Alex Acero, Hsiao-Wuen Hon. 2001. *Spoken Language Processing: a Guide to Theory, Algorithm, and System Development*. Prentice Hall PTR.
- Wayne Iba and Pat Langley. 1992. Induction of One-Level Decision Trees. In *ML92: Proceedings of the Ninth International Conference on Machine Learning, Aberdeen, Scotland, 1–3 July 1992*, pages 233–240.
- Sathiya S. Keerthi, S. K. Shevade, C. Bhattacharyya, and K. R. K. Murthy. 2001. Improvements to Platt’s SMO Algorithm for SVM Classifier Design. *Neural Computation*, 13(3):637–649.
- Katrin Kirchhoff and Sonia Parandekar. 2001. Multi-stream Statistical N-gram Modeling with Application to Automatic Language Identification. In *INTERSPEECH*, number 1, pages 803–806.
- Peter Ladefoged. 2001. *A Course in Phonetics*. volume 53. Cengage learning.
- Lori F. Lamel and Jean-Luc Gauvain. 1994. Language Identification Using Phone-Based Acoustic Likelihoods. In *Proceedings of ICASSP '94. IEEE International Conference on Acoustics, Speech and Signal Processing*, volume i, page I/293–I/296 vol.1.
- Niels Landwehr, Mark Hall, and Eibe Frank. 2005. Logistic Model Trees. *Machine Learning*, 59(1-2):161–205.
- Haizhou Li Haizhou Li, Bin Ma Bin Ma, and Chin-Hui Lee Chin-Hui Lee. 2007. A Vector Space Modeling Approach to Spoken Language Identification. *IEEE Transactions on Audio, Speech, and Language Processing*, 15(1):271–284.
- Richard P. Lippmann. 1997. Speech Recognition by Machines and Humans. *Speech Communication*, 22(1):1–15.
- Ignacio Lopez-Moreno, Javier Gonzalez-Dominguez, Oldrich Plchot, David Martínez, Joaquin Gonzalez-Rodriguez, and Pedro Moreno. 2014. Automatic Language Identification Using Deep Neural Networks. *Icassp*:0–4.

- David Martinez, Eduardo Lleida, Alfonso Ortega, and Antonio Miguel. 2013. Prosodic Features and Formant Modeling for an Ivector-based Language Recognition System. In *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, pages 6847–6851. IEEE.
- David Martínez, Oldřich Plchot, Lukáš Burget, Ondřej Glembek, and Pavel Matějka. 2011. Language Recognition in iVectors Space. *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*(August):861–864.
- Pavel Matejka, Petr Schwarz, Jan Cernocký, and Pavel Chytil. 2005. Tuning Phonotactic Language Identification System. Technical Report 4.
- Yeshwant K. Muthusamy, Etienne Barnard, and Ronald a. Cole. 1994. Reviewing Automatic Language Identification. *IEEE Signal Processing Magazine*, 11(4):33–41.
- Yeshwant Kumar Muthusamy, Kay M Berkling, T Arai, Ronald a Cole, and E Barnard. 1993. A Comparison of Approaches to Automatic Language Identification Using Telephone Speech. In *3rd European Conference on Speech Communication and Technology*, volume 2, pages 1307–1310.
- Thangavelu Nagarajan and H. A. Murthy. 2004. Language identification using parallel syllable-like unit recognition. In *2004 IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 1, pages I–401. IEEE.
- François Pellegrino and Régine Andre-Obrecht. 2000. Automatic language identification: an alternative approach to phonetic modelling. *Signal Processing*, 80(7):1231–1244.
- John C. Platt. 1998. Sequential Minimal Optimization: a Fast Algorithm for Training Support Vector Machines. *Advances in Kernel MethodsSupport Vector Learning*, 208:1–21.
- Olivier Pourret. 2008. *Bayesian Networks: a Practical Guide to Applications*. volume 73. John Wiley & Sons.
- John Ross Quinlan. 1993. *Programs for Machine Learning*. volume 240. Elsevier.
- John Ross Quinlan. 2014. *C4. 5: Programs for Machine Learning*. Elsevier.
- Lawrence R. Rabiner. 1989. Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition. *Proceedings of the IEEE*, 77(2):257–286.
- Itahashi Shuichi and Du Liang. 1995. Language Identification Based on Speech Fundamental Frequency. In *4th European Conference on Speech Communication and Technology*, volume 2, pages 1359–1362.
- Stanley S. Stevens. 1937. A Scale for the Measurement of the Psychological Magnitude Pitch. *The Journal of the Acoustical Society of America*, 8(3):185.
- Stanley S. Stevens. 1939. The Relation of Pitch to the Duration of a Tone. *The Journal of the Acoustical Society of America*, 10(3):255.
- Marc Sumner, Eibe Frank, and Mark Hall. 2005. Speeding up Logistic Model Tree Induction. In *Knowledge Discovery in Databases: PKDD 2005*, volume 3721, pages 675–683. Springer.
- Ann E. Thyme-Gobbel and S. E. Hutchins. 1996. On Using Prosodic Cues in Automatic Language Identification. In *Proceeding of Fourth International Conference on Spoken Language Processing. ICSLP '96*, volume 3, pages 1768–1772.
- Ingo R. Titze and Daniel W. Martin. 1998. Principles of Voice Production. *The Journal of the Acoustical Society of America*, 104(3):1148.
- Pedro A. Torres-Carrasquillo, Douglas A. Reynolds, and J. R. Deller. 2002. Language Identification Using Gaussian Mixture Model Tokenization. In *2002 IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 1, pages I–757–I–760.
- Geoffrey I. Webb. 2000. MultiBoosting: a Technique for Combining Boosting and Wagging. *Machine Learning*, 40(2):159–196.
- Yiming Yang and Jan O Pedersen. 1997. A Comparative Study on Feature Selection in Text Categorization. In *Proceedings of the Fourteenth International Conference on Machine Learning*, volume 97, pages 412–420.
- Yeshwant K. Muthusamy, Ronald A. Cole, and Beatrice T. Oshika. 1992. The OGI Multi-language Telephone Speech Corpus. In *Proceedings of the International Conference on Spoken Language Proceedings*

(*ICSLP, 現 INTERSPEECH*), volume 92, pages 895–898. Citeseer.

Steve Young, Gunnar Evermann, Dan Kershaw, Gareth Moore, Julian Odell, Dave Ollason, Valtcho Valtchev, and Phil Woodland. 2002. *The HTK book*. volume 3. Entropic Cambridge Research Laboratory Cambridge.

Stephen A Zahorian and Hongbing Hu. 2008a. YAAPT Pitch Tracking MATLAB Function. *The Journal of the Acoustical Society of America*, 123:4559–4571.

Stephen A. Zahorian and Hongbing Hu. 2008b. A Spectral/Temporal Method for Robust Fundamental Frequency Tracking. *The Journal of the Acoustical Society of America*, 123(6):4559–4571.

Marc A. Zissman. 1996. Comparison of Four Approaches to Automatic Language Identification of Telephone Speech. *IEEE Transactions on Speech and Audio Processing*, 4(1):31–44, January.

Marc A. Zissman and E. Singer. 1994. Automatic Language Identification of Telephone Speech Messages using Phoneme Recognition and N-gram Modeling. In *Proceedings of ICASSP '94. IEEE International Conference on Acoustics, Speech and Signal Processing*, volume i, pages I–305.

## The Syntactic and Semantic Analysis of *Hěn X* Constructions in Spoken Corpora

**Lai, Huei-ling**  
National Chengchi  
University  
Taipei, Taiwan

[hllai@nccu.edu.tw](mailto:hllai@nccu.edu.tw)

**Chen, Yen-ju**  
National Taiwan Normal  
University  
Taipei, Taiwan

[born95101078@gmail.com](mailto:born95101078@gmail.com)

**Hsu, Shao-chun**  
National Chengchi  
University  
Taipei, Taiwan

[shaochun@nccu.edu.tw](mailto:shaochun@nccu.edu.tw)

### Abstract

To understand how daily usages can shape the gradual changes of both *hěn*, a prototypical intensifier in Mandarin Chinese, and the construction *hěn X*, the study aims to investigate the syntactic and semantic behaviors of *hěn X* constructions in spoken corpora. The conversational data from the NCCU Corpus of Spoken Chinese and a Taiwan Public Television show *Bring Up Parents* are extracted and analyzed, focusing in particular on the syntactic categories of *X*, the grammaticalization of *hěn*, and the lexicalization of *hěn X*. Several findings are found. First, the syntactic and semantic distributions of the data from both corpora are quite consistent. While adjectives and stative verbs still claim the majority of *X*, new categories are discovered, showing host expansion of *X*. In addition to words, phrases and clauses can play the role of *X*. The increase of the flexibility and complexity of *X* demonstrates the gradual grammaticalization of *hěn*. Moreover, some instances of *hěn X* can be used as a unit to modify other grammatical constituents, showing a certain degree of lexicalization. When *hěn X* is fused as a unit, *hěn* is obligatory, not only indicating a degree but also highlighting the characteristics of *X*. The analysis shows that the nature of spoken materials enhances the subjectivity of *hěn X*. The findings of *hěn X* in spoken corpora can be applied to linguistic studies and Mandarin teaching.

### 1 Background

*Hěn X* constructions, often employed in both spontaneous speeches and written texts, have undergone syntactic and semantic changes. In addition to modifying common adjectives and verbs, the degree adverb *hěn* collocates with various types of words. While many studies have discussed the history of *hěn*, the development of its degree-specifying function and the expansion of *X*, in general, the main claim is that *hěn*, as a prototypical intensifier in Mandarin Chinese, has grammaticalized into a grammatical marker in conjunction with its gradual loss of lexical meaning but its gaining of subjective evaluation (Chui, 2000; Lin, 2009; Tseng, 2010; Bai and Zhao, 2007; Chen and Tsai, 2008; Liu and Chang, 2012). Among previous studies, few have discussed *hěn X* constructions in spoken data although they are used more and more frequently in daily conversations with *hěn* indicating a higher degree than normal states and with *X* expanding to various syntactic categories. The usages of *hěn X* constructions in spoken corpora deserve further exploration. To understand how daily usages can shape the gradual changes of *hěn X*, the study aims to investigate the syntactic and semantic behaviors of *hěn X* constructions in two different spoken corpora, focusing in particular on examining the syntactic categories of *X*, the grammaticalization of *hěn*, and the lexicalization of *hěn X*.

## 2 Grammaticalization and Lexicalization

Brinton and Traugott (2005:96-99) emphasize the highly interactive relation between grammaticalization and lexicalization in language change. Lexicalization refers to a word formation process in which a new lexical item is produced with its structural and semantic properties not completely derivable from the components of the word formation pattern. The output of such a process forms a gradient continuum of complexity, ranging from fixed or idiomatic phrases (L1), to compounds and derived forms (L2), and to lexical simplexes and idiosyncratic fossilized forms (L3). The degree of lexicalization within a word increases along with the loss of its grammatical and semantic element features, and lexicalization processes form a gradient continuum by the three levels of lexicality L1, L2, and L3. Grammaticalization, on the other hand, refers to a process whereby lexical items or constructions are used to serve a grammatical function in certain linguistic contexts, and become more grammatical by obtaining more grammatical functions and expanding their host-classes. Grammaticalization processes also form a gradient continuum on a scale of grammaticality G1, G2, and G3. Brinton and Traugott (2005) point out the differences and similarities of the two processes. Lexicalization integrates existing forms to serve as members of a major category, but grammaticalization involves decategorization of forms from major categories to minor ones to serve grammatical functions. However, both processes involve a decrease in syntactic or semantic compositionality and an increase in fusion. The analysis of this study indicates that both processes are involved in *hěn* constructions as will be shown below.

## 3 Methodology

### 3.1 Data

The data are taken from The NCCU Corpus of Spoken Chinese (Chui and Lai, 2008) and the TV interview show *Bring Up Parents* (爸媽冏很大) from Taiwan Public Television Service Foundation. The NCCU Corpus of Spoken Chinese, an online open access spoken data, consists of around 9 hours of 27 Mandarin daily conversations with two or three Mandarin-speaking adults. *Bring Up*

*Parents* is a TV program containing interviews and conversations of both parents and their sons or daughters. The episodes from July to December in 2013 were selected. The First and Eighth episodes in every month were extracted, totaling 12 episodes of 12 hours. In total, 805 tokens of the NCCU corpus, and 870 tokens of the TV show will be examined.

### 3.2 Data coding

The tokens of *hěn X* are coded regarding the syntactic categories of *X*, the number of words of *X*, the grammatical function of *hěn X*, and the meaning of *hěn X*. The procedure is shown below.

**(A) Syntactic category:** Analyzing the syntax category of *X* as NOUN, VERB, ADJECTIVE, ADVERB, PRONOUN, or PREPOSITION.

**(B) Word number:** Counting word number of *X* following *hěn*.

**(C) Grammatical function:** Indicating the grammatical functions of *hěn X* as SUBJECT, PREDICATE, OBJECT, ATTRIBUTIVE, ADVERBIAL, or COMPLEMENT.

**(D) Semantic function:** Observing the contexts of *hěn X*, and analyzing the meaning.

## 4 Results and Discussion

### 4.1 Results

The result is shown in Appendix 1. The distributions are quite similar in the two spoken corpora. Regarding syntactic categories, adjectives and stative verbs claim the majority of *X*. However, new categories such as nouns, relation verbs, modal verbs, adverbs, prepositions, and pronouns are found, showing host expansion of *X*. Regarding the syntactic functions of *hěn X*, serving as a predicate displays a major part in the distribution, and all complements are modal complements. Serving as subjects or objects are rare. The distributions of attributives and adverbials are alike. The only difference is that restrictive attributives are used more frequently in TV shows, and mostly is the lexicalized form *hěnduō* (很多 *hen-many*; 'frequent'). Regarding word numbers, *X* is found to contain one or two words in majority. When *X* contains three words, the string is usually a relational verb. The syntactic behaviors and meanings of *hěn X* vary in actual usages. When

used to modify adjectives and stative verbs, *hěn* objectively indicates the degree of shapes or quality. Consider the examples from (1) to (3). Long or thin in shape, hard or soft in property, and sour, bitter or stinky in senses may correspond to temper, cultural, life and emotions. Notice that an interesting feature of *hěn* is that sometimes it seems to be semantically bleached, becoming an obligatory marker. In these three examples, the three predicates cannot stand alone without *hěn*; however, there is no intensification present in the sentences. The construction of *hěn X* is lexicalized to some extent with metaphorical meaning extended from the whole construction.

- (1) 有時候老公會問的**很細**，然後我就會覺得很煩

*Yǒushíhòu lǎogōng huì wèn de hěnxì ránhòu wǒ jiù huì juéde hěnfán* (sometimes-husband-will-ask-COMP-very detailed-then-I-will-feel-very annoyed)

‘When my husband asks for too many details, I will feel very annoyed.’

- (2) 生物老師**很硬**，考的生物非常難。

*Shēngwù lǎoshī hěnyìng kǎo de shēngwù fēicháng nán* (biology-teacher-very tough -test-biology-very difficult)

‘Biology teachers are very tough, often giving students difficult tests.’

- (3) 結婚**很苦**，碰到很多波折。

*jiéhūn hěnkǔ pèngdào hěnduō bōzhé* (marriage-very bitter-bump into -many-frustrations)

‘Marriage is bitter; I bump into many frustrations.’

Due to its property-modifying function, *hěn* can collocate with *X* denoting state or property. Thus, the host *X* can further extend to nouns, action verbs, relational verbs, modal verbs, pronouns, adverbs, and prepositions, which could not be modified before. The meaning is metaphorical with *hěn X* lexicalized as a fused unit. When modifying a noun, *hěn* will trigger the appropriate semantic property contingent to linguistic contexts. For instance, in the case of *hěntǔ* (很土 *hen*-earth; ‘out of fashion’), the concrete property of earth changes into projecting the purpose and function of the

substance. Also, a metaphorical meaning is observed in the compound. *Hěn* highlights the fixed and invariable property of earth to metaphorically express a pejorative extended meaning--out of fashion. Cases that carry similar metaphorical meanings are *hěnbāgǔ* (很八股 *hen*-stereotyped; ‘hackneyed’) or *hěnkǒuhào* (很口號 *hen*-slogan; ‘like a slogan’).

Grammaticalization of *hěn* occurs, with *hěn* extending to modify a noun denoting abstract orientation, as in the following example:

- (4) 天母...房子都打**很**下面，然後下面都是停車場...。

*Tiānmǔ fángzi dōu dǎ hěnxìamiàn ránhòu xiàmiàn dōu shì tíngchēchǎng* (Tianmu-house-all-build-very low -then-down-all-is-garage)

‘Houses in Tianmu are built farther under the ground for parking garages....’

The chunk *hěnxìamiàn* denotes farther down the ground, and *hěn* emphasizes not only the degree of orientation but also the speaker’s subjective evaluation of the situation.

The categories of *X* can also expand to relational verbs, modal verbs and action verbs. One interesting example has to do with the co-occurrence of *hěn* with *you X* construction to emphasize a high degree above the average. For example, in (5), *hěnyǒutónggǎn* (很有同感 *hen*-have-same feeling ‘feel the same way’) is to emphasize the speaker’s feeling and thinking. And in (6) *hěnyǒugǎnqíng* (很有感情 *hen*-have-feelings; ‘have deep feelings’) is a grammatical unit to express the speaker’s emotions. Thus, *hěn* is employed to denote a high degree associated with *you* constructions.

- (5) 我**很有同感**，我很贊同他說的。

*wǒ hěnyǒutónggǎn wǒ hěn zàntóng tā shuō de* (I-hen-have-the same feeling-I-hen-agree-what he has said)

‘I feel exactly the same way. I totally agree with what he has said.’

- (6) 他沒有對我**很有感情**。

*tā méiyǒu duì wǒ hěnyǒugǎnqíng* (he-not-duit-me-hen-have- feeling)

‘He didn’t have deep feelings toward me.’

When modal verbs indicating obligation collocate with *hěn*, the construction denotes speakers' subjectivity toward judging facts and emotions, and the collocation with *hěn* strengthens the speakers' subjectivity. The case *hěn bù yīnggāi* (很 不 應 該 *hen-not-should*; 'really shouldn't') can illustrate. The case *hěn huì zhǔ* (很 會 煮 *hen-able-cook*; 'really good at cooking') shows the speaker's evaluation of someone's talent in cook whereas *hěn huì tánliànài de* (很 會 談 戀 愛 *hen-able-romance*; 'good at handling romantic relationships') carries the speaker's evaluation of someone being good at romantic relationships. Furthermore, *hěn* can modify lexicalized action verb phrases, as in *hěn chīlì* (很 吃 力 *hen-eat-strength*; 'very laborious'), *hěn xiàrén* (很 嚇 人 *hen-scared*; 'very scary'), or *hěn jiànyì* (很 建 議 *hen-recommend*; 'highly recommend') and *hěn shuǎshuài* (很 耍 帥 *hen-show-handsome*; 'look very cool'). Notice that in these cases *X* is getting more and more lexicalized, and that *hěn* is obligatory, revealing its grammatical function from grammaticalization.

The host classes of *X* keep expanding to pronouns as *hěn* further grammaticalizes. For example, in (7) and (8), speakers express euphemism by employing *hěnnàge* (很 那 個 *hen-that-CL*; 'really-you know-bad'). These two examples show that *nàge*, a deictic expression, indicates abstract events and states. The usage of *hěnnàge* euphemistically expresses speakers' negative thoughts toward the states. The co-occurrence of *hěn* and pronouns carries strong subjectivity since what the deictic pronoun *nàge* refers to can only be understood by contexts.

(7) 自殺真的很那個...

*Zìshā zhēnde hěnnàge* (commit suicide-really-  
hen-that CL)

'It is really bad to commit suicide.'

(8) 我覺得考試去看電影很那個。

*Wǒ juéde kǎoshì qù kàn diànyǐng hěnnàge* (I-  
feel-exam-go-see movies -hen-that CL)

'I think it's not very good to see movies before the exam.'

Finally, *X* can even expand to include prepositions. Prepositions are function words indicating relations. In examples (9) and (10), *hěn* is a

grammatical marker, strengthening the degree of the head specified by the predicate; it serves to express the relation between the speaker and the role modified by the preposition.

(9) 很替你感到難過

*Hěn tì nǐ gǎn-dào nánguò* (hen-for-you-feel-  
sorry)

'feeling really sorry for you'

(10) 很向鼻子靠近

*Hěn xiàng bízi kào jìn* (hen-toward-nose-  
approach)

'approaching really toward the nose'

## 4.2 Grammaticalization and Lexicalization of *hěn X*

The discussion has shown that *hěn* is a degree adverb, intensifying the degree of its head specified by *X*. Due to the nature of spoken materials, *hěn* is further grammaticalized with *X* being further expanded to include longer strings of words and more complex syntactic structures. While serving various grammatical functions like predicates, attributives, adverbials, or complements, some instances of *hěn X* such as *hěnhǎo* (很好 *hen-good*; 'very good'), *hěnnán* (很難 *hen-hard*; 'hard to...'), *hěnduō* (很多 *hen-many* 'frequent'), and *hěنشǎo* (很少 *hen-little* 'little') are lexicalized as fused units ready to modify other constituents.

Interestingly, the process of lexicalization continues as in cases such as *hěnhǎo X*, *hěnnán X*, *hěnduō X*, *hěnyǒu X*, and *hěnxǎng X*. Such findings indicate that grammaticalization and lexicalization are highly interrelated processes. Owing to the frequent use of *hěn* with *hǎo* 'good', *nán* 'difficult', and *duō* 'many', and *shǎo* 'few', their word boundaries gradually diminish, producing a new semantic unit. For example, *hěnnán* can indicate either evaluation of possibility or the degree of difficulty as in *hěnnánshuō* (很難說 *hen-hard-say* 'hard to say') and *hěnnán yǒukòng* (很難有空 *hen-hard-available* 'hard to be available'). While *duō* and *shǎo* represent amounts, *hěnduō* or *hěنشǎo* indicate frequency when qualifying abstracts or states as in *bāng hěnduō máng* (幫很多忙 *ban-hen-many-help* 'often help a lot') and *hěنشǎo tán zhèjiànshì* (很少談這件事 *hen-little-discuss-this-matter* 'seldom discuss this

matter’). The meaning of *hěnhǎo* has shifted from evaluating good quality to assess possibility *hěnhǎozhuī* (很好追 *hen-good-chase* ‘easy to hit on’).

The empirical findings of this study support the integration of grammaticalization and lexicalization proposed by Brinton and Traugott (2005). These two processes, motivated by speakers’ needs in interaction, undergo instantaneous changes and reanalysis. Language forms are repeatedly conducted by interlocutors, and gradually become fossilized. These gradual changes are dynamic with indeterminacy, revealing speakers’ subjective attitudes in daily usages. The subjectivity of the development of *hěn X* is justified as *hěn X* constructions mostly serve grammatical functions as predicates, descriptive attributives adverbials, or modal complements. The following two figures summarize the analysis of *hěn X* constructions. Figure 1 shows the gradual expansion of X from more prototypical categories like adjectives and verbs to less prototypical ones like nouns, pronouns, and prepositions.

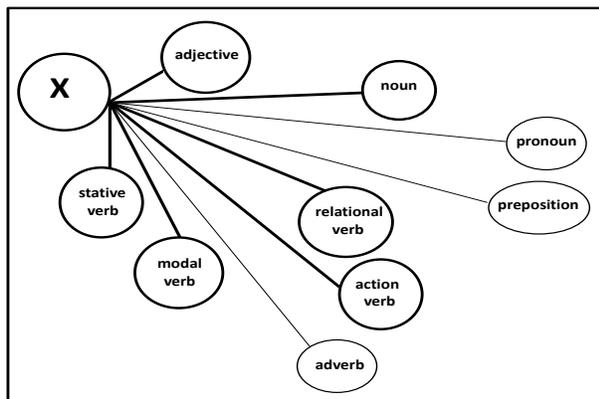


Figure 1. The expansion of X

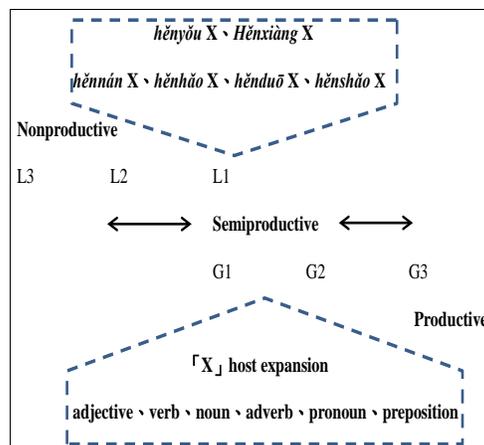


Figure 2. Synchronic clines of lexicality and grammaticality

Figure 2 indicates that while the host X is expanded, *hěn* is becoming more and more grammaticalized into a grammatical marker. Some *hěn X* constructions such as *hěnhǎo X*, *hěnnán X*, *hěnduō X*, and *hěnhǎo X*, have lexicalized into a unit due to its frequent usage in spoken data. These cases have also developed their evaluative and subjective meanings in the contexts.

### 5 Conclusion

Grammatical and semantic changes happen due to speakers’ needs. This current work inspects the structural and semantic changes of *hěn* as an intensifier, as well as the syntactic and semantic behaviors of *hěn X* constructions in spoken corpora through quantitative and qualitative methods. The conversational data from the NCCU Corpus of Spoken Chinese and a Taiwan Public Television show *Bring Up Parents* are extracted and analyzed. Several findings are found. First, the syntactic and semantic distributions of the data from both corpora are quite consistent. Due to the nature of spoken materials, X reveals host expansion, expanding to new categories including nouns, adverbs, prepositions and pronouns other than the prototypical adjectives and stative verbs. It can also include not only words but also phrases and clauses. The increase of the flexibility and complexity of X demonstrates further grammaticalization of *hěn*. When X keeps on expanding to other syntactic categories, *hěn X* is developing toward the direction of grammaticality with an increase of its productivity (Briton and

Traugott, 2005). However, some instances of *hěn X* become lexicalized units serving to modify other grammatical constituents. When *hěn X* is fused as a unit, *hěn* becomes an obligatory grammatical marker, expressing a higher degree than normal and at the same time highlighting the features denoted by its host. And such fused constructions are developing toward the direction of lexicality with a decrease of productivity (Briton and Traugott, 2005). The dynamic and interactive nature of conversations enhances the subjectivity of *hěn X*, in contingent with the integration of the processes of grammaticalization and lexicalization. The findings of *hěn X* in spoken corpora can be applied to linguistic studies and Mandarin teaching.

## References

- Brinton, Laurel and Traugott, Elizabeth Closs. 2005. *Lexicalization and Language Change*. Cambridge University Press, Cambridge, UK.
- Chen, Ying and Tsai, Zheng. 2008. *Adverbs Hěn and Zhěn*. 現代漢語虛詞研究與對外漢語教學, *Xian dai han yu xu ci yan jiu yu dui wai han yu jiao xue*, 2, 14-27.
- Chui, Kawai. 2000. Morphologization of the Degree Adverb HEN. *Language and Linguistics*, 1(1):45-59.
- Chui, Kawai, and Lai, Huei-ling. 2008. The NCCU Spoken Corpora: Mandarin, Hakka, and Southern Min. *Taiwan Journal of Linguistics*, 6(2): 119-144.
- Hong, Jia-Fei and Huang, Chu-Ren. 2006. Using Chinese Gigaword Corpus and Chinese Word Sketch in Linguistic Research. In *Proceedings of the 20th Pacific Asian Conference on Language, Information and Computation (PALIC'20)*, Wuhan, China.
- Lin, Wen-yin. 2009. The use of *Hěn* in Modern Chinese. In International Workshop on Theory and Practice in Mandarin Chinese Instruction 2009, Taipei, Taiwan.
- Liu, Mei-Chun, and Chang, Chun. 2012. The Degree-evaluative Construction: Grammaticalization in Constructionalization. In Xing Zhiqun (ed.) *Newest Trends in the Study of Grammaticalization and Lexicalization in Chinese*, 115-148. Mouton de Gruyter, Berlin.
- Pai, Hsiao-hung, and Chao, Wei. 2007. 漢語虛詞十五講, *Han-yü hsü-tz'u 15 chiang* [Function words in Chinese]. Peking University Publisher, Beijing.
- Tseng, Wei. 2010. Degree highlighting in spontaneous speech. *Journal of Shaoxing University*, 30(3), 67-72.

**Appendix: Syntactic Category and Grammatical Function Distributions in the Corpora**

Syntactic category Source		Noun		Verb		Adjective		Pronoun		Adverb		Preposition	
		Corpus	Show	Corpus	Show	Corpus	Show	Corpus	Show	Corpus	Show	Corpus	Show
Grammatical function													
Subject		0	0	0	1	6	0	0	0	0	0	0	0
		0.00%	0.00%	0.00%	0.11%	0.74%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
Predicate		9	7	112	156	443	329	4	0	1	1	1	2
		1.11%	0.80%	13.91%	17.93%	55.03%	37.81%	0.50%	0.00%	0.12%	0.11%	0.12%	0.22%
Object		0	0	3	0	1	4	0	0	0	0	0	0
		0.00%	0.00%	0.37%	0.00%	0.12%	0.45%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
Attributive	Restrictive	0	0	0	0	27	105	0	0	0	0	0	0
		0.00%	0.00%	0.00%	0.00%	3.35%	12.06%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
	Descriptive	1	2	5	4	56	87	0	0	0	0	0	2
		0.12%	0.22%	0.62%	0.45%	6.95%	10.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.22%
Adverbial	Restrictive	0	0	1	3	1	2	0	0	0	0	0	0
		0.00%	0.00%	0.12%	0.34%	0.12%	0.22%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
	Descriptive	1	1	2	9	31	65	0	0	16	7	0	0
		0.12%	0.11%	0.24%	1.03%	3.85%	7.47%	0.00%	0.00%	1.98%	0.80%	0.00%	0.00%
Compliment	Resultative	0	0	0	0	0	0	0	0	0	0	0	0
		0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
	Directional	0	0	0	0	0	0	0	0	0	0	0	0
		0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
	Potential	0	0	0	0	0	0	0	0	0	0	0	0
		0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
	Modal	1	0	1	1	82	82	0	0	0	0	0	0
		0.12%	0.00%	0.12%	0.11%	10.18%	9.42%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
	Quantity	0	0	0	0	0	0	0	0	0	0	0	0
		0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
<b>Total</b>		<b>12</b>	<b>10</b>	<b>124</b>	<b>174</b>	<b>647</b>	<b>674</b>	<b>4</b>	<b>0</b>	<b>17</b>	<b>8</b>	<b>1</b>	<b>4</b>
<b>Percentage</b>		<b>1.49%</b>	<b>1.15%</b>	<b>15.40%</b>	<b>20.00%</b>	<b>80.37%</b>	<b>77.47%</b>	<b>0.50%</b>	<b>0.00%</b>	<b>2.11%</b>	<b>0.80%</b>	<b>0.12%</b>	<b>0.45%</b>

# Methods and Tool for Constructing Phonetically-Balanced Materials for Speech Perception Testing: A Development of Thai Sentence-Length Materials

## Adirek Munthuli

Department of Electrical and Computer Engineering, Faculty of Engineering, Thammasat University, Thailand  
5310450027@student.tu.ac.th

## Charturong Tantibundhit

Department of Electrical and Computer Engineering, Faculty of Engineering and Center of Excellence in Intelligent Informatics, Speech and Language Technology, and Service Innovation (CILS) Thammasat University, Thailand  
tchartur@engr.tu.ac.th

## Chutamane Onsuwan

Department of Linguistics, Faculty of Liberal Arts and Center of Excellence in Intelligent Informatics, Speech and Language Technology, and Service Innovation (CILS) Thammasat University, Thailand  
consuwan@tu.ac.th

## Krit Kosawat

National Electronics and Computer Technology Center (NECTEC), National Science and Technology Development Agency (NSTDA), Thailand  
krit.kosawat@nectec.or.th

## Abstract

Phonemic content is one of many important criteria in a development of any kind of speech testing materials. In this paper, we explain a procedure and tool we created in the process of constructing phonetically-balanced (PB) sentence-length materials for Thai, as an assessment for speech reception thresholds. Our procedure includes establishing criteria, preselecting sentences, creating pool of replacement words, determining phonemic distribution, and constructing sentences. Importantly, a tool is created to determine whether set of words or sentences are phonetically balanced. Once the phoneme distribution and the set of words with transcription are specified, the tool efficiently computes phoneme occurrences among words or sentences (within a set) and can be used to manipulate words to achieve goal in

phonetically balanced (PB). To show how this is accomplished, two sentence sets are constructed and evaluated by native speakers. The procedure and tool have characteristics that make them potentially useful in other applications and can be applied to other languages.

**Keywords:** Thai, sentence-length material construction, phonetically balanced speech materials

## 1 Introduction

It is well-established that an assessment technique for evaluating an individual's hearing sensitivity based on pure-tone audiometry alone does not truly reflect the individual's speech understanding (Bilger et al., 1984; Egan, 1948). Importantly, measuring of speech intelligibility could be obtained by counting number of correct responses from speech testing materials, e.g., phonetically-balanced (PB) monosyllabic words, polysyllabic words, and sentences (Egan, 1948).

For the Thai language, there are a few existing speech materials for intelligibility test, some of these were developed using monosyllabic word lists, e.g., RAMA.SD1, RAMA.SD2 (Komalarajun, 1979), and TU PB'14 (Munthuli et al., 2014) while some using phrase or sentence materials, e.g., Ramathibodi Synthetic Sentence Identification (RAMA.SSI), which contains Thai artificial sentences (with no real meaning) (Wissawapaisal, 2002), and “PB and PD sentences”, which are long stretches of phrases and sentences derived from Thai continuous speech corpus for an evaluation of automatic speech recognition system (Wutiw WATCHAI et al., 2002). However, a majority of speech testing methods using sentence materials requires that the sentences are representative of the real communication system, which includes many factors such as meaning, context, rhythm, etc. (Egan, 1948). It is quite clear that the existing Thai sentence materials would not satisfactorily meet this requirement.

Sentence speech materials have been created in many languages, e.g., Dutch (Plomp and Mimpen, 1979), Mandarin Chinese (Fu et al., 2011), German (Kollmeier and Wesselkamp, 1997). For English (American), the most widely used are Speech Perception in Noise test (SPIN) (Kalikow et al., 1977) and Hearing in Noise Test (HINT) (Nilsson et al., 1994). SPIN is a test for measuring speech intelligibility at fixed S/N ratio, but it was found to have variability in terms of sentence difficulty (Kalikow et al., 1977). Therefore, HINT was designed and developed as a Hearing in Noise Test, composed of lists of sentences, which are shown to have no significant difference in terms of difficulty. Among those, different strategies were used (but no specific tool had been mentioned) to construct the phonetically balanced materials. It should be noted that those materials were created to obtain similar phoneme distributions among sentence sets (see Phonemic content in Table 1) rather than to reflect the true phonemic distributions of the language. Our approach tries to achieve both ends by using a semi-automatic tool. A fully automated tool of this type would be ideal, but would require other crucial components such as a language model. A list of important characteristics of SPIN and HINT are given in Table 1.

Due to the lack of Thai ‘natural’ sentence materials for speech perception testing, and

especially those for assessing hearing-impaired individuals. Our goal is to construct Thai phonetically balanced sentence-length materials for assessing speech reception thresholds. In this paper, we describe the methods and tool for constructing a subset of these meaningful sentences.

	SPIN (1977)	HINT (1994)
Sentence length	6-8 syllables	6-9 syllables
Number of lists	8	25
Number of sentences per list	50	10
Measurement	Speech intelligibility (count only ‘keyword’ at the last monosyllabic noun of the sentence)	Speech intelligibility (count every word of the sentence) and sentence speech reception threshold (sSRT)
Phonemic content	Balanced within class of phonemes from Dewey’s written corpus	Phonemically balanced of 43 phoneme sounds among lists
Others	Low predictability (LP) and high predictability (HP) sentences	Sentence difficulty: 1-grade reading level

Table 1: Important characteristics of SPIN and HINT tests.

## 2 Establishing Criteria

The first requirement in constructing PB sentence materials is phonetic/phonemic balance. Other common criteria include word familiarity, naturalness, sentence length, homogeneity, test-retest reliability, and inter-list difficulty (Bilger et al., 1984). Our approach is to incorporate most of the above criteria. However, in this paper, our focus is on the initial phase, which is designing lists of natural sentences with phonetic balance, equal length, and familiar words. The next phase, testing and evaluating, not discussed here, will be to ensure homogeneity, test-retest reliability, and inter-list difficulty.

Our PB sentence lists are based on phoneme distribution of Thai speech LOTUS-CELL2.0 (LT-CS) corpus (Section 3.2). To minimize effect of subject’s different language

background, we opt for familiar words. This is carried out by selecting words and sentences, which match desired phonemic content, from children's textbooks and stories, (Thai Children Stories, 1990; Ministry of Education, 1986; Sripaiwan, 1994; Sangworasin, 2003). In terms of sentence length, we follow SPIN (Kalikow et al., 1977), HINT (Nilsson et al. 1994), and RAMA.SSI (Wissawapaisal, 2002) and limit each sentence to six to eight syllables with no words greater than two syllables long. The PB sentences will compose of 10 lists, each with 10 sentences.

In addition, to address a question of whether different levels of predictability affect sentence intelligibility (Kalikow et al., 1977), in all five lists will be created to fit the 'low' predictability status and another five the 'high' predictability. However, degrees of predictability are beyond the scope of our developed tool, and are determined by semantics and overall sentence contexts. (see Sections 4 and 5).

### 3 Procedure and Concept Design for Tool

For SPIN (Kalikow et al., 1977) and HINT (Nilsson et al., 1994), pre-selection of sentences were carried out prior to phonemic distribution analysis and matching. HINT sentences were selected from Bamford-Kowal-Bench (BKB) corpus. SPIN sentences were constructed by generating sets of 'low' predictability and 'high' predictability sentences and manipulated key words (monosyllable nouns) in sentence final position by determining their semantics link to preceding words in the sentence. The key words were drawn from Thorndike-Lorge corpus. Consequently, for HINT, there is 68% (of 252 sentences) where phonemes are off  $\pm 1$  from the target phonemes (Nilsson et al., 1994).

We found their approach quite difficult to achieve for Thai sentences as there are 4 phoneme types (initials, vowel, finals, and lexical tones) to account for. Therefore, we have taken a slightly different approach by starting with pre-selection of sentences in the same fashion, but the sentences will be further modified by replacing and reconstructing some words in sentences until it yields desired phonemic contents as described in Section 3.1.

Kalikow et al. (1977) asserted that recognition of keywords in sentence is based on

familiarity of word. Therefore, for our lists, we have to make certain that the selected words (candidates) are familiar words in the language. We do so, by selecting words from children's textbooks and stories. In addition, to estimate frequency of word occurrences, we utilize the largest available Thai written corpus InterBEST (Kosawat et al., 2009).

Most importantly, our PB word candidates are considered to be as phonetic balanced as possible, i.e., less than 10% difference from targeted phoneme distribution.

#### 3.1 Preselecting Sentences and Pool of Replacement Words

The first step to create PB sentences is based on preselection of sentences. All sentences from a collection of 89 children's stories (Thai Children Stories, 1990) are analyzed and only simple sentences, (i.e., subject-verb-(adverb), subject-verb-complement/object), are kept. These result in 313 sentences in total. Then, each sentence is transcribed and its phonemic distribution of initials, finals, vowels, and tones are tallied. Attempts are made to group a set of 10 sentences in to a list (10 lists in all) such that the phoneme distributions are as close to the ones shown in Tables 2-5 as possible.

However, from a limited number of simple sentences (313 sentences) that were preselected, the best outcome we could obtain was 10 lists of useable PB sentences with very low off-target from the desired phoneme distributions.

Therefore, we propose an additional step, which is to modify our preselected sentences by replacing and reconstructing some words using a pool of replacement words so that it finally yields ten mutually exclusive groups of 10 sentences that match the desired phoneme distributions.

Our pool of replacement words came from the collection of 89 children's stories (Thai Children Stories, 1990) and word corpora based on three children's textbooks (Ministry of Education, 1986; Sripaiwan, 1994; Sangworasin 2003).

#### 3.2 Phonemic Content

In this section, phoneme frequency occurrence and its distribution (ranking) derived from written and spoken Thai corpora (Kosawat et al., 2009;

Chotimongkol et al., 2009; Chotimongkol et al., 2010) are discussed (Munthuli et al., 2015). More generally, InterBEST, which is one of the Thai largest written corpora, is composed of 12 text genres with approximately nine million words. LOTUS-CELL2.0 is a collection of telephone conversation recordings of 50 hours long, where data were transcribed according to different speaking styles: formal style (LT-FS) and causal speech style (LT-CS) (Chotimongkol et al., 2010). LOTUS-BN is a Thai television broadcast news recordings of 100 hours long. Munthuli et al. (2015) show phoneme distribution from InterBEST, LOTUS-CELL2.0 (LT-FS and LT-CS) and LOTUS-BN. Among the written and two spoken corpora, there are notable differences (largely due to lexical differences and phonetic variations in conversational speech) in terms of frequency occurrence and the distribution for initial consonants, vowels, final consonants (but not for lexical tones) (Munthuli et al., 2015). In addition, many existing speech testing materials (e.g., HINT) favored the use of spoken corpus (Nilsson et al., 1994). Therefore, our approach here is to employ the phoneme frequency occurrence and distribution derived from causal speech style (LT-CS). The next step is to modify our preselected sentences by replacing and reconstructing some words (using the pool of replacement words in Section 3.1) so that it finally yields ten mutually exclusive groups of 10 sentences that match the desired phoneme distributions as shown in Tables 2-5 as much as possible.

### 3.3 Selecting and Replacing Words

A tool is developed to facilitate the process at which the preselected sentences are modified by replacing and reconstructing some words using a pool of replacement words so that it finally yields ten mutually exclusive groups of 10 sentences that match the desired phoneme distributions. The steps involved are as follows:

1. Consider target number of phoneme occurrence of all 65 phonemes shown in Tables 2-5 (29 initials, 21 vowels, 10 finals, and 5 tones) that are required for construction of PB sentences.
2. Start with construction of PB sentences of List 1. Consider all combinations of the preselected sentences; choose 10 sentences

( ${}^{313}C_{10}$ ). Then, the selected 10 sentences will be transcribed and the resulting phonemes are tallied.

3. For each case of the selected 10 sentences, calculate absolute difference between numbers of occurrences of Step 1 and Step 2 for each phoneme. Then, calculate percentage of summation of absolute differences for all phonemes.
4. Select the best combination of 10 simple sentences (6-8 syllables per sentence), where the sentences provide the lowest percentage of summation of absolute differences for all phonemes. After this selection, these 10 sentences will be removed from the list of preselected sentences.
5. Consider the phonemes in Step 4, where numbers of occurrences are higher than target numbers in Step 1. These phonemes will be among the first phonemes to be removed.
6. Consider all words from the selected sentences in Step 4, which compose of phonemes (initials, finals, vowels, or tones) in Step 5. These words will be removed in order based on which one has a higher number of exceeding phonemes per syllable. In case of tie, the one with lower word frequency of occurrences based on InterBEST corpus (Kosawat et al., 2009) has higher priority to be removed. It should be noted that a two-syllabic word has a higher priority than a monosyllabic word. Then, update number of occurrences of all phonemes.
7. Repeat Step 6 until no exceeding phoneme available.
8. Now, all phonemes have numbers of occurrences below target numbers. Then, insert a new word from a pool of replacement words. Words with higher frequency of occurrences will have higher priority. Then, update number of occurrences of all phonemes.
9. Repeat Step 8 until numbers of occurrences of all phonemes of preselected sentences have absolute error less than 10%, i.e., any phoneme in any group of initials, finals, vowels, and tones can be

out of target at most 2 times and 6 times in total.

10. Use words in the replacement pool to construct 10 sentences, where each sentence is a simple sentence composed of seven syllables.

Repeat Steps 1 to 10 to construct PB sentences for Lists 2 to 5.

#### 4 Tool

Graphical user interface is developed to facilitate insertion or removal of words by considering each list of PB sentences one by one.

Figures 1-2 show asterisks on the chart. Each of which signifies a target number of occurrences of any phoneme. Bar refers to current number of occurrences of any phoneme, where positive/negative number signifies that number of occurrences is higher/lower than a target number of occurrences of that phoneme.

Figure 1 shows an example of the PB sentence constructing process with the tool starting with Step 1 described earlier in Section 3.3. Here, we

show construction of sentences in list I (low predictability sentences). After Steps 2 to 3 are performed, select the best combination of 10 sentences stated in Step 4. Then, each sentence (one by one) is put in the tool as shown in Fig. 1.

After Steps 5 to 7 are performed, insert words from pool of replacement words. Figures 1-2 show words ranked in ascending order based on frequency of occurrences (from InterBEST corpus).

Figure 2 shows phoneme distributions after performing Step 9. Then, Step 10 is performed and 10 PB sentences with low predictability are shown in Table 6. As another example, we use the tool to construct 10 PB sentences with high predictability as shown in Table 7.

It should be noted that degrees of predictability are beyond the scope of this tool, and are determined by semantics and overall sentence contexts. At this stage, the tool users are expected to make several attempts in word selecting and replacing to achieve desired level of predictability, which could be later evaluated (see Section 5).

	b	tɕ	tɕ <sup>h</sup>	d	f	h	j	k	k <sup>h</sup>	k <sup>h</sup> r	k <sup>h</sup> w	kl	kr	kw	l	m	n	ŋ	p	p <sup>h</sup>	p <sup>h</sup> l	p <sup>h</sup> r	pr	r	s	t	t <sup>h</sup>	w	ʔ
List 1	2	2	2	3	0	2	3	5	6	0	0	1	0	0	7	5	5	1	3	3	0	0	0	1	5	2	4	3	5
List 2	2	3	2	3	0	2	2	5	7	0	0	0	0	0	8	5	4	1	3	3	0	1	0	0	5	3	4	2	5
List 3	2	3	2	2	1	2	2	5	7	0	0	0	0	0	7	6	4	1	3	3	0	0	1	0	5	2	5	2	5
List 4	2	2	2	3	0	2	3	5	6	0	0	0	1	0	7	5	5	1	3	3	0	0	0	1	5	2	4	3	5
List 5	2	3	1	3	1	2	2	5	6	0	0	0	0	1	7	5	5	0	3	3	0	0	0	1	5	3	4	2	6
List 6	2	2	2	3	0	2	2	6	6	1	0	0	0	0	7	5	5	1	3	3	0	0	0	1	5	2	4	3	5
List 7	2	2	2	3	1	1	3	5	6	0	1	0	0	0	7	5	5	1	3	3	0	0	0	1	5	2	4	3	5
List 8	1	3	2	3	0	2	2	6	6	0	0	0	0	1	7	5	4	1	4	2	0	0	0	1	5	3	4	2	6
List 9	2	2	2	3	1	1	3	5	6	0	1	0	0	0	7	5	5	1	3	3	0	0	0	1	5	2	4	3	5
List 10	1	3	2	3	0	2	2	6	6	0	0	0	0	0	8	5	4	1	3	3	1	0	0	1	4	3	4	2	6

Table 2: Initial consonant occurrences across ten sentence lists.

	ɔ	ɔɔ	a	a:	e	e:	i	i:	ia:	o	o:	ɤ	ɤ:	u	u:	ua:	ur	ur:	uaa:	ε	ε:
List 1	1	7	22	11	2	2	2	5	1	3	1	0	1	2	2	1	1	2	1	0	3
List 2	1	7	21	12	2	1	3	4	2	2	1	0	2	2	1	1	1	2	1	1	3
List 3	1	7	21	12	2	2	2	5	1	2	1	0	2	2	2	1	1	2	1	0	3
List 4	1	6	22	12	2	1	3	4	2	2	1	0	2	1	2	1	1	2	1	1	3
List 5	1	7	21	12	2	1	3	4	2	2	1	0	2	2	1	2	1	1	1	1	3
List 6	0	7	22	12	2	1	2	5	1	3	1	0	2	1	2	1	1	2	1	0	4
List 7	1	7	21	12	2	1	3	4	2	2	1	1	2	1	2	1	1	2	1	0	3
List 8	0	7	22	12	1	2	2	5	1	3	1	0	2	2	1	2	1	1	1	1	3
List 9	1	7	21	12	2	1	3	4	2	2	1	1	1	2	2	1	1	2	1	0	3
List 10	0	7	22	12	1	2	2	5	1	3	1	0	2	2	1	1	2	1	1	1	3

Table 3: Vowel occurrences across ten sentence lists.

	j	k	m	n	ŋ	p	t	w	‘x’	ø
List 1	9	3	3	10	6	3	4	4	9	19
List 2	9	3	3	9	6	4	3	4	10	19
List 3	9	3	3	9	6	4	3	4	10	19
List 4	8	3	4	10	6	3	4	4	9	19
List 5	9	3	3	9	7	3	4	3	10	19
List 6	8	3	4	9	6	4	3	4	9	20
List 7	9	3	3	9	7	3	4	3	10	19
List 8	8	3	4	9	6	4	3	4	9	20
List 9	9	3	3	9	6	4	4	3	10	19
List 10	8	3	3	10	6	3	4	4	9	20

Table 4: Final consonant occurrences across ten sentence lists.

	ˉ	ˊ	ˋ	ˌ	ˍ
List 1	23	14	16	11	6
List 2	22	15	16	11	6
List 3	23	14	16	11	6
List 4	22	15	16	10	7
List 5	23	15	15	11	6
List 6	23	14	16	11	6
List 7	23	15	15	11	6
List 8	22	15	16	11	6
List 9	23	9	16	11	6
List 10	22	10	16	11	6

Table 5: Lexical tone occurrences across ten sentence lists.

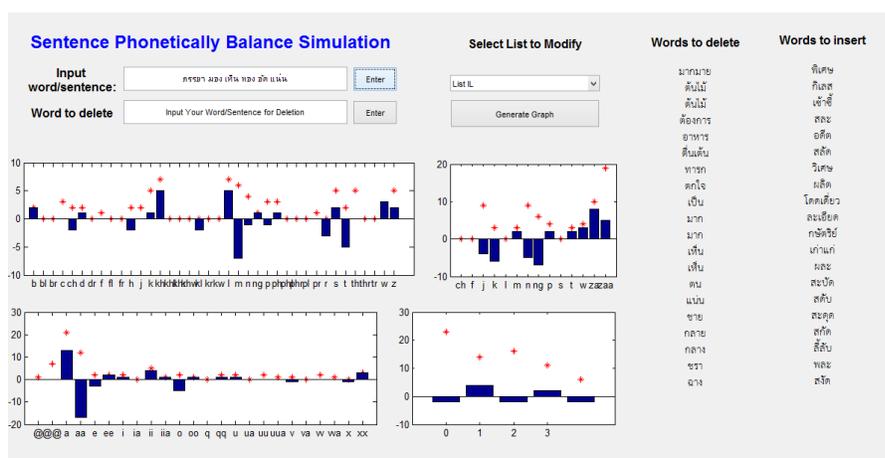


Figure 1: Simulation of tool in process of subtraction in a set of preselected sentences that has minimum absolute of summation errors. It should be noted that c is ɕ, ch is ɕʰ, kh is kʰ, khl is kʰl, khr is kʰr, khw is kʰw, ng is ŋ, ph is pʰ, phl is pʰl, phr is pʰr, th is tʰ, thl is tʰl, za is ‘x’, zaa is ø, @ is ɔ, @@ is ɔ:, aa is a:, ee is e:, ii is i:, iia is ia:, oo is o:, q is ɣ, qq is ɣ:, uu is u:, uua is ua:, v is ʋ, vv is ʋ:, vva is ʋa:, x is ɛ, xx is ɛ:, 0 is mid tone, 1 is low tone, 2 is falling tone, 3 is high tone and 4 is rising tone.

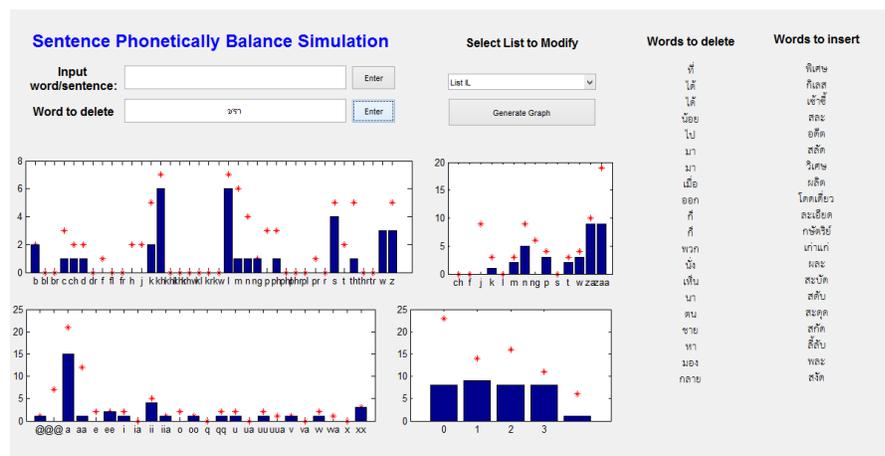


Figure 2: Simulation of tool in process of removing words in a set of sentences which has exceeding phonemes.

### 5 Preliminary Output and Evaluation

Tables 6 and 7 show two lists of Thai PB sentences that are successfully constructed using the tool. Importantly, differences from the target phoneme distributions are lower than 10% for each type of phoneme (initial consonant, vowel, final consonant, and lexical tone).

Another important step, which we incorporate into our procedure, is to analyze and evaluate our attempts in word selecting to achieve desired level of predictability. In so doing, we statistically compare evaluation responses from Thai raters and determine whether they rate ‘low’ and ‘high’ predictability sentences differently

We combine 20 sentences (constructed ‘high’ (Type 4) and ‘low’ (Type 2) predictability sentences) listed in Tables 6 and 7 with 20 sentences drawn from the list of our preselected sentences (Section 3.1). Ten of the twenty preselected sentences could potentially be considered as highly predictable (Type 3) and the other 10 with low predictability (Type 1). Twenty Thai adult participants are asked to rate each sentence in five-point scales (5 = very high predictability, 1 = very low predictability). Mean and average rating score of four types of sentences are given in Table 8.

หญิงสาว แบบเบาะ ก็ สร้าง ชาติ ‘young lady’ ‘baby’ ‘also’ ‘build’ ‘nation’ A baby lady also builds a nation. [jǐŋ sǎ:w bē:əbòx kô:ə sá:ŋ tɕʰá:t]
เป็ด ดื้อ กลับ หา กำไร ได้ ‘duck’ ‘stubborn’ ‘become to’ ‘find’ ‘profit’ ‘get’ A stubborn duck is making a profit. [pèt d ũ:ə klàp há:ə kāmraj dā:j]
พวกตน เหวี่ยง ลิ้น ออก ไป ‘we’ ‘fling’ ‘tongue’ ‘out’ ‘go’ We fling the tongue out. [pʰú:a:k tōn wia:ŋ lín ?ò:k pāj]
ภรรยา มองเห็น คอ ขยับ ‘wife’ ‘see’ ‘neck’ ‘move’ A wife sees the neck moving. [pʰānráx jā:ə mō:ŋ hěn kʰō:ə kʰà x jàp]
ชาย ใจดำ เล้าโลม งู ‘man’ ‘black-hearted’ ‘fondle’ ‘snake’ Black-hearted man fondles a snake. [tɕʰā:j tɕəjdām lǎwlō:m ŋū:ə]
แพะ สามารถ ทะเลาะ กับ เวลา

‘goat’ ‘can’ ‘quarrel’ ‘with’ ‘time’ A goat can quarrel with time. [pʰéx sǎ:əmə:t tʰá x lóx kàp wē:əlā:ə]
ท่าน ลอย ไป แก่คั้น มา ‘you’ ‘float’ ‘go’ ‘revenge’ ‘come’ You float to get revenge. [tʰân lō:j pāj kɛ:kʰé:n mā:ə]
เจ้าของ ก็ นั่ง เข้าซี้ อีก ‘owner’ ‘also’ ‘sit’ ‘importune’ ‘again’ An owner sits and importunes again. [tɕəwkʰō:ŋ kô:ə nâŋ sáwsí:ə ?ì:k]
หนู มี เวทมนตร์ ใน ขณะ นี้ ‘mouse’ ‘has’ ‘magic’ ‘in’ ‘while’ ‘this’ A mouse is currently having magic power. [nū:ə mī:ə wē:tmōn nāj kʰà x nà x ní:ə]
ต้นไม้ ทอง ขึ้น อยู่ ที่อื่น ละ ‘tree’ ‘gold’ ‘grow’ ‘at’ ‘elsewhere’ ‘already’ A golden tree already grew up elsewhere. [tōnmá:j tʰō:ŋ kʰūn jù:ə tʰī:ə ?ù:n lá x]

Table 6: Example of a set of ‘low’ predictability sentences (constructed sentences) (‘x’ signifies an ending of any short-vowel syllables with no final consonant whereas ‘ə’ a syllable with long vowel with no final consonant).

ข้า น้อย พับ เสื้อผ้า รอ เป็น วัน ‘I’ ‘fold’ ‘cloth’ ‘wait’ ‘is’ ‘day’ I folded clothes for a day while I am waiting. [kʰā:ə nō:j pʰáp sʰu:a:əpʰā:ə rō:ə pēn wān]
ยาย เล่า วิธี แกะสลัก ‘grandmother’ ‘describe’ ‘method’ ‘carving’ Grandmother describes how to carve. [jā:j lāw wí x tʰī:ə kè x sà x làk]
ท่าน หิว เพิ่ม ขึ้น ไป อีก ‘you’ ‘hungry’ ‘increase’ ‘up’ ‘go’ ‘more’ You get hungrier. [tʰân hīw pʰî:m kʰūn pāj ?ì:k]
เขา ต้อง ขอบคุณ อาจารย์ มาก ‘He’ ‘must’ ‘thankful’ ‘professor’ ‘many’ He must be very thankful to professor. [kʰǎw tōŋ kʰò:pkʰūn ?ā:ətəā:n mā:k]
เจ้า โอ้อวด แม้ ยัง สงสัย ‘you’ ‘show off’ ‘even’ ‘still’ ‘doubt’ You are showing off even if you still have a doubt. [tɕəw ?ò:ə?u:a:t mé:ə jāŋ sǒŋsǎj]
ขณะนี้ น้อง ไม่ เฮฮา ‘while’ ‘this’ ‘brother/sister’ ‘not’ ‘joyful’ Brother/Sister is not joyful at this moment. [kʰà x nà x ní:ə nō:ŋ mâj hē:əhā:ə]
เธอ น้อยใจ ก็ ทะเลาะ อีก

‘she’ ‘feel slight’ ‘then’ ‘quarrel’ ‘again’ If she feels slighted, quarrel will begin again. [t <sup>h</sup> ɰ:ø nɔːjtɕəj kɔːø t <sup>h</sup> á x lɔː x ʔiːk]
บัณฑิต แต่ละ คน มี ชื่อเสียง ‘graduate’ ‘each’ ‘person’ ‘has’ ‘famous’ Each Graduate is famous. [bāndit tɛːø lá x k <sup>h</sup> ɔn mī:ø tɕ <sup>h</sup> ɦ:ø s̩iɑːŋ]
บุตรหลาน ดูแล ไม่ ง่าย ‘children’ ‘take care’ ‘not’ ‘easy’ [Taking care of children is not easy. bʉt lǎ:n dū:ølɛːø mâj ŋá:j]
ฉัน กำลัง ตาม เก็บ กุหลาบ ‘I’ ‘being’ ‘follow’ ‘pick’ ‘rose’ I am picking roses. [tɕ <sup>h</sup> ǎn kām̩lāŋ tā:m kɛp kù x là:p]

Table 7: Example of a set of ‘high’ predictability sentence (constructed sentences) (‘x’ signifies an ending of any short-vowel syllables with no final consonant whereas ‘ø’ a syllable with long vowel with no final consonant).

	Mean	Standard deviation
Type 1: ‘Low’ predictability (Preselected)	1.72	0.48
Type 2: ‘Low’ predictability (Constructed)	1.86	0.80
Type 3: ‘High’ predictability (Preselected)	4.48	0.30
Type 4: ‘High’ predictability (Constructed)	3.20	0.80

Table 8: Mean and average rating score of four types of sentences.

We perform ANOVA to test differences between high predictability and low predictability sentence types and use multiple comparisons to check whether each pair is statistically significant as shown in Figure 3. As expected, results show that significant differences are found in pairs of Types 1 and 3; and Types 2 and 4. An important point to be taken here is that levels of predictability could be estimated and later evaluated by native speakers (but this is beyond the scope of the developed tool).

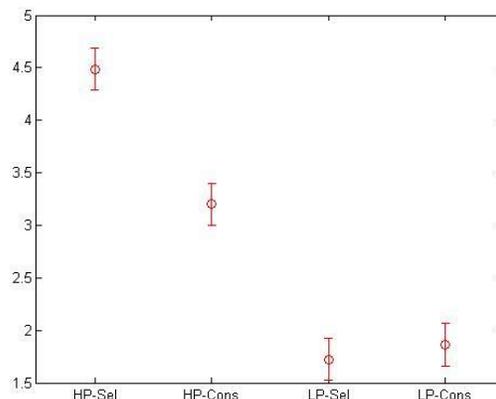


Figure 3: Multiple comparisons between 4 types of sentences. It should be noted that LP-Sel, LP-Cons, HP-Sel, and HP-Cons are referred to Type 1, Type 2, Type 3, and Type 4, respectively.

## 6 Discussion and Future Direction

We believe that we have successfully proposed and outlined procedure as well as constructed an efficient tool for constructing PB sentence sets. Importantly, a main advantage of our proposed procedure and tool is that it is easy to administer and create sets of words that are close to the desired distribution. As previously mentioned, a fully automated tool of this type would be ideal, but would require other crucial components such as a language model and other information associated with each word (e.g., part of speech).

The procedure and tool outlined here have characteristics that make them potentially useful in other applications and can be applied to other languages, but will certainly require a language specific set of data (i.e., phoneme distribution and language-specific grapheme-to-phoneme software).

## Acknowledgments

This work was supported by Thailand Graduate Institute of Science and Technology (TGIST), NSTDA (TGIST 01-57-020) to the first author. Special thanks to Thanaporn Anansiripinyo for her help with Thai-to-English gloss and translation.

## References

R. C. Bilger, J. M. Nuetzel, W. M. Rabinowitz, and C. Rzeczkowski. 1984. Standardization of a Test of Speech Perception in Noise. *Journal of Speech, Language, and Hearing Research*, 27(1): 32–48.

- A. Chotimongkol, K. Saykhum, P. Chotrakool, N. Thatphithakkul, C. Wutiwiwatchai. 2009. LOTUS-BN: A Thai Broadcast News Corpus and Its Research Applications. *Proceedings of Oriental-COCOSDA*, Xinjiang, China.
- A. Chotimongkol, N. Thatphithakkul, S. Purodakananda, C. Wutiwiwatchai, P. Chotrakool, C. Hansakunbuntheung, A. Suchato, and P. Boonpramuk. 2010. The Development of the Large Thai Telephone Speech Corpus: LOTUS-Cell 2.0. *Proceedings of Oriental-COCOSDA*, Kathmandu, Nepal.
- J. P. Egan. 1948. Articulation Testing Methods. *Laryngoscope*, 58(9): 955–991.
- Q. Fu, M. Zhu, and X. Wang. 2011. Development and Validation of the Mandarin Speech Perception Test. *Journal of the Acoustical Society of America*, 129(6): EL267–273.
- D. N. Kalikow, K. N. Stevens, and L. L. Elliot. 1977. Development of a Test of Speech Intelligibility in Noise using Sentence Materials with Controlled Word Predictability. *Journal of the Acoustical Society of America*, 61(5): 1337–51.
- B. Kollmeier and M. Wesselkamp. 1997. Development and Evaluation of a German Sentence Test for Objective and Subjective Speech Intelligibility Assessment. *Journal of the Acoustical Society of America*, 102(4): 2412–21.
- S. Komalarajun. 1979. *Development of Thai Speech Discrimination Materials*. Master's thesis, Department of Communication Disorders, Faculty of Graduate Studies, Mahidol University, Bangkok, Thailand.
- K. Kosawat, M. Boriboon, P. Chotrakool, A. Chotimongkol, S. Klaithin, S. Kongyoung, K. Kriengkiet, S. Phaholphinyo, S. Purodakananda, T. Thanakulwarapas, and C. Wutiwiwatchai. 2009. BEST 2009: Thai Word Segmentation Software Contest. *Proceedings of 8<sup>th</sup> International Symposium on Natural Language Processing*, Bangkok, Thailand: 83–88.
- Ministry of Education. 1986. *Basic words for teaching Thai language primary education*. Department of Curriculum and Instruction Development, Bangkok, Thailand.
- A. Munthuli, P. Sirimujalin, C. Tantibundhit, K. Kosawat, and C. Onsuwan. 2014. Constructing Thai Phonetically Balanced Word Recognition Test in Speech Audiometry through Large Written Corpora. *Proceedings of 17<sup>th</sup> Oriental Chapter of COCOSDA*, Phuket, Thailand.
- A. Munthuli, C. Tantibundhit, C. Onsuwan, K. Kosawat, and C. Wutiwiwatchai. 2015. Frequency of Occurrence of Phonemes and Syllables in Thai: Analysis of Spoken and Written Corpora. *Proceedings of 18<sup>th</sup> International Congress of Phonetic Sciences*, Glasgow, Scotland.
- M. Nilsson, S. D. Soli, and J. A. Sullivan. 1994. Development of the Hearing in Noise Test for the Measurement of Speech Reception Thresholds in Quiet and in Noise. *Journal of the Acoustical Society of America*, 95(2): 1085–99.
- R. Plomp and A. M. Mimpen. 1979. Improving the Reliability of Testing the Speech Reception Threshold for Sentences. *Audiology*, 18(1): 43–52.
- R. Sangworasin. 2003. *Education of Kindergarten's Vocabulary 4 -5 years old in Bangkok*. Bachelor's Thesis, Department of Education, Primary Education, Faculty of Education, Chulalongkorn University, Bangkok, Thailand.
- R. Sripaiwan. 1994. *Thai textbooks set "Mana Manee Piti Chujai"*. Department of Curriculum and Instruction Development, Ministry of Education, Bangkok, Thailand.
- Thai Children Stories. 1990. Retrieved from <http://www.nithan.in.th/>.
- A. Thangthai, C. Hansakunbuntheung, R. Siricharoenchai, and C. Wutiwiwatchai. 2006. Automatic Syllable Pattern Induction in Statistical Thai Text-to-phone Transcription. *Proceedings of Interspeech*, Pittsburgh, PA.
- N. J. Versfeld, L. Daalder, J. M. Festen, and T. Houtgast. 2000. Method for the Selection of Sentence Materials for Efficient Measurement of the Speech Reception Threshold. *Journal of the Acoustical Society of America*, 107(3): 1671–84.
- W. Wissawapaisal. 2002. *Development of the Thai Synthetic Sentence Identification Test*. Master's thesis, Department of Communication Disorders, Faculty of Graduate Studies, Mahidol University, Bangkok, Thailand.
- C. Wutiwiwatchai, P. Cotsomrong, S. Suebvisai, and S. Kanokphara. 2002. Phonetically Distributed Continuous Speech Corpus for Thai Language. *Proceedings of 3<sup>rd</sup> International Conference on Language Resources and Evaluation*, Las Palmas, Canary Islands, Spain: 869–872.

# Graph Theoretic Features of the Adult Mental Lexicon Predict Language Production in Mandarin: Clustering Coefficient

**Karl David Neergaard**

The Hong Kong Polytechnic  
University  
136 Shanghai St. 7/F  
Hong Kong

karl.neergaard@connect.polyu.hk

**Chu-Ren Huang**

The Hong Kong Polytechnic  
University  
11 Yuk Choi Rd.  
Hong Kong

churen.huang@polyu.edu.hk

## Abstract

Graph theory has recently been used to explore the mathematical structure of the mental lexicon. In this study we tested the influence of graph measures on Mandarin speech production. Thirty-six native Mandarin-speaking adults took part in a shadowing task containing 194 monosyllabic words, 94 of which consisted of 3 phonemes and were the items under analysis. Linear mixed effect modeling revealed that clustering coefficient (C) predicted spoken production of Mandarin monosyllabic words, while network degree, in this case its phonological neighborhood density (PND) failed to account for lexical processing. High C resulted in shorter reaction times, contrary to evidence in English. While these findings suggest that lexical processing is affected by the network structure of the mental lexicon, they also suggest that language specific traits lead to differing behavioral outcomes. While PND can be understood as the underlying lattice for which a similarity network is created, lexical selection is not affected by only a target word's neighbors but instead the level of interconnectivity of words (C) within the network.

## 1 Introduction

Graph theory is currently an active tool within the language sciences. Networks constructed from the semantic knowledge of children have shown typical versus disordered development (Beckage et al., 2011) and helped to explain the growth of vocabulary (Hills et al., 2009). The network structure of phonological networks has been found to influence children's productive vocabulary development and failed lexical retrieval in adults (Vitevitch et al., 2014). The new methodology coming to form involves the combination of graph theoretic models and psycholinguistic tasks, allowing for a view into the lexicon to examine language processing according to structural relations.

The manner in which a phonological network is constructed is through what is known as phonological neighborhood density (PND), which is a similarity metric that involves the addition, deletion or substitution of a single phoneme (Vitevitch, 2008). Thus, in the network, words (nodes) are connected (edges) to one another based on their sound similarity. Words that are connected via this similarity are known as neighbors and give us the network feature known as degree ( $k$ ). In the psycholinguistic literature PND has been extensively investigated. It has been shown to influence word recognition (Luce and Pisoni, 1998), production (Sadat et al., 2014), and word learning (Storkel et al., 2006) to just name a few.

Once the network is built, other measures are then available, such as each node's clustering coef-

ficient (C). C is the number of triangles made in relation to a given node. In terms of the mental lexicon, this presents us with a measure of how interconnected a word's neighbors are with each other. It has been illustrated with an English lexicon that PND and C are not equivalent measures in that they do not correlate with each other (Chan and Vitevitch, 2009). The role of C has been examined in word recognition (Chan and Vitevitch, 2009; Yates, 2013), and picture naming (Chan and Vitevitch, 2010), allowing for the tentative statement that, at least for English speakers, low C words are produced faster and more accurately than high C words.

While research into the network features of the mental lexicon has advanced rapidly, there has been an inordinate stress upon European languages, specifically English. Mandarin, to date has no evidence of either a PND or C effect on language processing, despite several attempts (Myers and Tsay, 2005; Tsai, 2007). One reason for such a disparaging lack might lie in the complexity of the Mandarin mental lexicon, specifically the role that tone plays. Indeed, Vitevitch and Stamer (2006) propose that differences in processing found between languages are likely to be found due to the linguistic differences exhibited by many languages.

In comparison to English, Mandarin has a small syllable inventory (~400 without tone). This language specific feature might suggest a lexicon that would be more dense, leading to increased competition between neighbors. Tone however creates distance between what would be otherwise similar sounding words. Tone, in fact has been shown to be the initial guiding point for phonological manipulation (Neergaard & Huang, 2016; Weiner & Turnbull, 2015).

The purpose of the current study is to investigate the role of network characteristics in a tonal language through the implementation of an auditory shadowing task.

## 2 Methods

### 2.1 Participants

The current results come from the spoken production of Thirty-six native Mandarin speakers (Female: 20). One participant was excluded from the analysis due to misunderstanding the task instruc-

tions. None of the participants reported speech, hearing, or visual disorders.

### 2.2 Stimuli

The stimuli, recorded by a female native Mandarin speaker from the Beijing area, consisted of 193 Mandarin monosyllabic words. All stimuli were 415ms in duration. Target stimuli, which can be seen in Appendix A, consisted of 94 words that were 3 phonemes in length. Filler words consisted of 99 monosyllabic words that contained 1, 2 and 4 phonemes. Filler words were used in the task so as to preclude the participants' ability to predict the structure of upcoming words. Presentation order

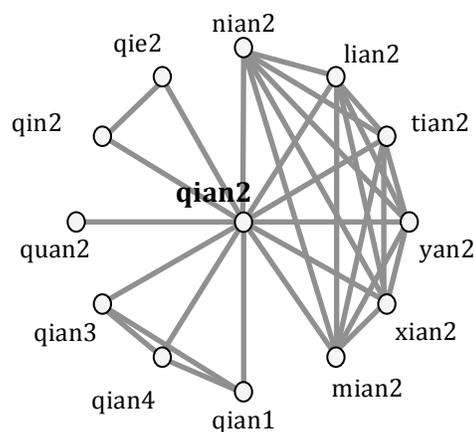


Figure 1. The word level network for qian2 /te'ien2/ 前 (PND: 12; C: .83)

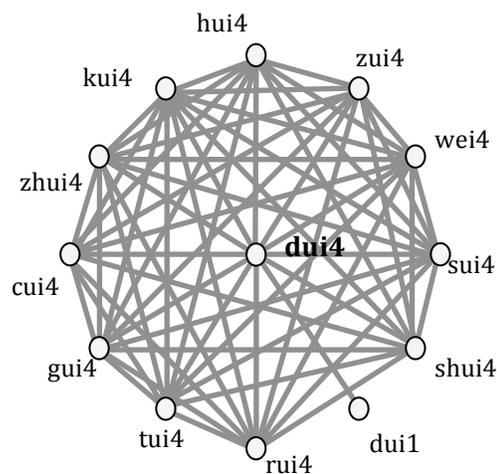


Figure 2. The word level network for dui4/ tuer4/对, (PND: 12; C: .29)

was pseudo-randomized so as not to allow for the serial presentation of words that began with the same onset or that had the same tone.

The stimulus words were selected from a database of movie subtitles (Subtlex-CH: Cai and Brysbaert, 2010). As is common amongst databases that provide calculations of PND and other lexical information (See Marian et al., 2012 for an in-depth discussion), a representative sample of orthographic words is chosen from either a dictionary or subtitle movie corpora. The current study calculated PND and four of the following word characteristics from the top 17 thousand entries of phonological words. The pinyin transcriptions of the Subtlex-CH database were made using the Lingua Sinica corpus (Chen et al., 1996). Phonological representations of spoken Mandarin were then taken from Neergaard and Huang (2016) according to the maximal syllable structure: CVVX plus tone.

The frequencies of homophonous words were summed together such that spoken word frequency (SWF) (M: 0.0271 per-million; SD: 0.0556 per-million), and homophone density (HD) could be calculated. HD (M: 5; SD: 4) was calculated based on the number of orthographic words that were used in the corpus per each phonological word. Neighborhood frequency (NF) (M: 18,1950; SD: 20,2318) was calculated from the combined frequency of a word's neighbors. C (M: 0.4065; SD: 0.1623) was calculated through the use of the network analysis tool, Gephi (Bastian et al., 2009). It should be noted that the correlation between PND (M: 14; SD: 5) and C within our stimuli set was low: 0.3. For an illustration of the difference between PND and C see Figures 1 and 2. Note that both represent words with equivalent densities.

### 2.3 Procedure

Participants were seated in a quiet room in front of a computer running experimental software, E-Prime 2.0 (Psychology Software Tools, 2012). They were instructed to repeat experimenter-provided auditory stimuli into a headset as quickly as possible. The onset of each trial was activated when a participant spoke via a PST Serial Response Box. They were given a practice set of 10 words.

Each trial consisted of the same sequence: “下一个词” (next word) was presented at the center of

the screen for 1000ms, followed by a blank screen and the onset of the target audio which changed either upon the onset of a participant's spoken response or a maximum of 3000ms, then finally a pause of 3000ms. The entire experiment took less than 15 minutes and was recorded on a second computer using Audacity 2.0.6.

Reaction times were measured offline using SayWhen (Jansen and Watter, 2008). The audio recordings were also used to transcribe the participants' spoken production by two native-Mandarin speaking volunteers. Incorrect responses were removed from the analysis, accounting for less than 6% of the data.

### 3 Results

Statistical analyses were done using linear mixed effect modeling (lmerTest in R). The first constructed model revealed that SWF ( $t = -2.462$ ;  $p = 0.014$ ) and C ( $t = -2.771$ ;  $p = 0.0056$ ) were predictors in the production of Mandarin monosyllabic words, while PND, NF, and HD were non-significant. In order to eliminate the effect of SWF on the other predictors, 96 trials, identified as outliers, were removed from the total of 3,177 trials. The removal of the outliers, which accounted for 3% of the total, limited the responses' reaction time to within 450 and 1000ms.

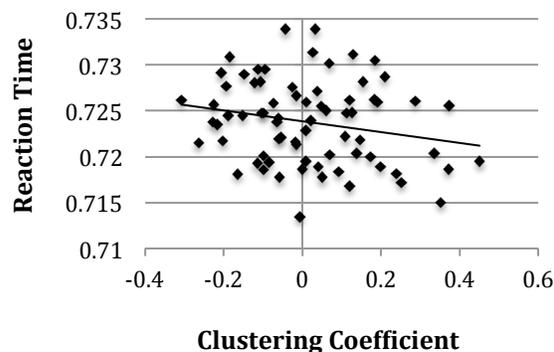


Figure 3. The effect of Mandarin clustering coefficient on reaction time

A second model was then created according to stepwise backwards model comparison. While the SWF effect disappeared, and none of the other predictors changed status, the effect of C remained through each successive

model iteration (Std. Error: 0.00820; df: 3037;  $t = -3.27$ ;  $p = 0.001$ ). Unique to this study, high C values resulted in shorter reaction times as can be seen in Figure 3.

#### 4 Conclusion

The present study is the first to find an influence of network measures on language production in a tonal language. Of particular note is the fact that the direction of the C effect is contrary to that of the English findings (Chan and Vitevitch, 2009, 2010). While the two prior studies implemented different tasks to what is currently featured, the direction was the same for English speakers: words with low C were produced faster and more accurately than words with high C. The present results, in contrast, suggest that the greater the interconnectivity of phonological words the less the competition for lexical selection.

One direction for further investigation is the role that network density plays across the Mandarin lexicon, specifically during development. If, like the present findings suggest, greater connectivity speeds processing, then this would imply the emergence of an adaptive trait learned through the acquisition of highly similar words. Such a language specific adaptation would have implications for vocabulary acquisition and possibly be of note for children with phonological delay (Gierut et al., 1999).

An alternative hypothesis is that a significant C effect concurrent with a null PND effect points to an error in the model's construction. There have been multiple proposals as to the segmentation of the Mandarin syllable (Duanmu, 2009). In the current study we examined a segmental approach with phonological tone. Future experimental designs would benefit from contrasting stimuli that have been calculated according to multiple segmentation schemas.

#### References

Duncan J. Watts, and Steven H. Strogatz. 1998. Collective dynamics of 'small-world' networks. *Nature* 393(6684):440-442.

Holly L Storkel, Jonna Armbruster, and Tiffany P. Hogan. 2006. Differentiating phonotactic probability and neighborhood density in adult word learning. *Journal of Speech, Language, and Hearing Research*, 49(6): 1175-1192.

James Myers, and Jane Tsay. 2005. The processing of phonological acceptability judgments. *Proceedings of Symposium on 90-92 NSC Projects*, pp. 26-45.

Jasmin Sadat, Clara D. Martin, Albert Costa, and F-Xavier Alario. 2014. Reconciling phonological neighborhood effects in speech production through single trial analysis. *Cognitive psychology*, 68:33-58.

Judith A. Gierut, Michele L. Morrisette, and Annette H. Champion. 1999. *Lexical constraints in phonological acquisition*. *Journal of Child Language*, 26:261-294.

Karl Neergaard, and Chu-Ren Huang. 2016. Phonological neighborhood density in a tonal language: Mandarin neighbor generation task. *90<sup>th</sup> Annual Meeting of the Linguistic Society of America*, Washington, DC.

Keh-jiann Chen, Chu-Ren Huang, Li-ping Chang, and Hui-Li Hsu. 1996. Sinica Corpus: Design Methodology for Balanced Corpora. *Proceeding of the 11<sup>th</sup> Pacific Asia Conference on Language, Information and Computation (PACLIC 11)*. pp. 167-176. Seoul, South Korea.

Kit Ying Chan, and Michael Vitevitch. 2009. The influence of the phonological neighborhood clustering coefficient on spoken word recognition. *Journal of Experimental Psychology: Human Perception and Performance*, 35:1934-1949.

Kit Ying Chan, and Michael Vitevitch. 2010. Network structure influences speech production. *Cognitive Science*, 34(4):685-697.

Mark Yates. 2013. How the clustering of phonological neighbors affects visual word recognition. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 39(5):1649.

Mathieu Bastian, Sebastien Heymann, and Mathieu Jacomy. 2009. Gephi: an open source software for exploring and manipulating networks. *The International AAAI Conference on Web and Social Media*, 8:361-362.

Michael S. Vitevitch. 2008. What can graph theory tell us about word learning and lexical retrieval?. *Journal of Speech, Language, and Hearing Research*. 51(2):408-422.

Michael S. Vitevitch, Kit Ying Chan, and Rutherford Goldstein. 2014. Insights into failed lexical retrieval

- from network science. *Cognitive Psychology*, 68:1-32.
- Michael S. Vitevitch, and Melissa K. Stamer. 2006. The curious case of competition in Spanish speech production. *Language and Cognitive Processes*, 21(6):760-770.
- Nicole Beckage, Linda Smith, and Thomas Hills. 2011. Small worlds and semantic network growth in typical and late talkers. *PLoS ONE*, 6(5):e19348.
- Paul A. Luce, and David B. Pisoni. 1998. Recognizing spoken words: The neighborhood activation model. *Ear and hearing*, 19(1):1.
- Pei-Tzu Tsai. 2007. The effects of phonological neighborhoods on spoken word recognition in Mandarin Chinese. Unpublished masters thesis.
- Peter A. Jansen, and Scott Watter. 2008. SayWhen: An automated method for high-accuracy speech onset detection. *Behavioral Research Methods*, 40(3):744-751.
- Psychology Software Tools, Inc. [E-Prime 2.0]. 2012. Pittsburgh, PA.
- Qing Cai, and Marc Brysbaert. 2010. Subtlex-CH: Chinese word and character frequency based on film subtitles. *PLoS ONE*, 5(6):e10729.
- San, Duanmu. 2011. Chinese syllable structure. *van Oostendorp, Marc/Ewen, Colin J./Hume, Elizabeth/Rice, Keren (Hg.): The Blackwell Companion to Phonology (5)*, 2754-2777.
- Seth Weiner, and Rory Turnbull. 2015. Constraints of Tones, Vowels and Consonants on Lexical Selection in Mandarin Chinese. *Language and Speech*:1-24.
- Thomas T Hills, Mounir Maouene, Josita Maouene, Adam Sheya, and Linda Smith. 2009. Longitudinal Analysis of Early Semantic Networks Preferential Attachment or Preferential Acquisition?. *Psychological Science*, 20(6):729-739.
- Viorica Marian, James Bartolotti, Sarah Chabal, and Anthony Shook. CLEARPOND: Cross-linguistic easy-access resource for phonological and orthographic neighborhood densities. *PloS ONE*, 7(8):e43230.

## Appendix A. Experimental Stimuli

Stimulus	Tone	SWF	HD	PND	NF	C
ban1	1	0.0077	8	20	83792	0.3895
bei3	3	0.0010	1	9	325390	0.4444
bing1	1	0.0025	2	14	165280	0.2857
bo1	1	0.0027	7	13	241279	0.6154
cai2	2	0.0370	4	15	214339	0.4000
cong2	2	0.0473	3	9	24192	0.4444
cuo4	4	0.0363	4	18	431772	0.6928
dai4	4	0.0385	10	21	896543	0.5333
die1	1	0.0003	1	13	46232	0.4744
fei2	2	0.0011	2	9	279495	0.5000
fen4	4	0.0159	6	15	39299	0.4571
feng1	1	0.0115	9	17	35947	0.5809
gai1	1	0.0342	1	19	108154	0.4327
gang1	1	0.0115	7	23	133498	0.3913
gao3	3	0.0185	3	21	151546	0.5190
gua4	4	0.0035	2	11	167761	0.2000

gun3	3	0.0040	2	12	33905	0.3030
hei1	1	0.0457	3	6	43405	0.4667
hen3	3	0.2011	3	11	24792	0.5273
hong2	2	0.0031	8	11	64484	0.3091
hou4	4	0.0226	4	16	450310	0.6583
hua2	2	0.0026	5	9	199436	0.2222
hun4	4	0.0035	2	16	227090	0.2917
huo2	2	0.0115	1	14	334170	0.4396
jia1	1	0.0332	13	12	106018	0.2121
jie1	1	0.0148	8	18	125483	0.3007
jin4	4	0.0247	10	11	104577	0.2000
jing3	3	0.0054	9	16	126033	0.3083
jue2	2	0.0026	13	5	16343	0.1000
jun1	1	0.0016	5	7	9095	0.1429
kua3	3	0.0004	1	9	8227	0.2222
kun4	4	0.0015	1	14	117829	0.3516
lao3	3	0.0177	4	22	1062897	0.4762
lie4	4	0.0016	8	12	112347	0.3333
lun2	2	0.0032	4	9	12206	0.2222
mai3	3	0.0158	1	15	48933	0.4476
mao1	1	0.0035	1	19	350311	0.5439
mei2	2	0.1617	13	9	129539	0.4444
men2	2	0.0089	2	12	248251	0.3485
min2	2	0.0005	4	11	45135	0.2182
ming2	2	0.0115	7	13	75796	0.3077
mo2	2	0.0035	8	12	176888	0.5909
nan2	2	0.0171	5	15	98327	0.5524
nong4	4	0.0136	1	9	83669	0.7778
pao4	4	0.0014	3	23	798622	0.4545
pei2	2	0.0035	5	9	274588	0.4444
pin1	1	0.0012	1	13	42567	0.2051
qia1	1	0.0004	1	9	60204	0.2222
qin2	2	0.0007	8	11	135499	0.2545
qing3	3	0.0306	2	14	56346	0.3516
que1	1	0.0007	2	7	22955	0.1905
qun2	2	0.0032	2	5	13142	0.2000
ran2	2	0.0007	3	14	260802	0.6044
rang4	4	0.1238	2	16	180346	0.7583
ren2	2	0.1950	5	11	24812	0.4000
reng1	1	0.0041	1	14	41806	0.8571
rou4	4	0.0025	1	15	123569	0.7429
ruo4	4	0.0022	5	17	467943	0.7794
san3	3	0.0002	2	22	58109	0.5974
sang1	1	0.0004	2	20	113667	0.5158
shan1	1	0.0034	12	19	81325	0.4152
shang4	4	0.1147	4	21	204382	0.4667

sheng3	3	0.0014	1	13	58529	0.3205
shua1	1	0.0006	2	12	299841	0.2424
shuo1	1	0.2270	1	13	25019	0.5897
song4	4	0.0133	5	10	84434	0.6444
tang3	3	0.0023	4	19	57176	0.3626
ting1	1	0.0555	3	13	76196	0.3077
wai4	4	0.0041	1	26	773415	0.3908
wan2	2	0.0244	6	20	201339	0.3421
wang4	4	0.0077	4	23	319519	0.4150
wei2	2	0.0493	13	13	289022	0.3077
wen4	4	0.0205	3	14	40995	0.5275
xia4	4	0.0557	4	11	115839	0.2182
xie2	2	0.0028	12	12	92857	0.1818
xin1	1	0.0265	9	12	69575	0.2576
xue2	2	0.0062	1	8	8282	0.1786
xun1	1	0.0004	5	7	34976	0.1429
yan3	3	0.0064	5	15	236670	0.3810
yang3	3	0.0036	4	22	317363	0.2944
yao4	4	0.2435	4	30	696392	0.3471
yong3	3	0.0010	12	14	381844	0.4066
you3	3	0.2896	6	22	175262	0.3463
yuan2	2	0.0121	15	7	24682	0.1905
zai3	3	0.0007	2	15	527024	0.4667
zao3	3	0.0129	5	21	193120	0.5190
zeng4	4	0.0001	3	13	42450	0.5897
zhan4	4	0.0132	9	22	245617	0.4286
zhen4	4	0.0024	7	15	469888	0.4476
zheng4	4	0.0207	5	16	315381	0.4000
zhong1	1	0.0382	6	14	56887	0.4176
zhua1	1	0.0084	1	10	34372	0.3111
zong3	3	0.0099	1	12	92516	0.5606
zun1	1	0.0004	5	8	6180	0.5357

Note: Stimulus words are presented in pinyin; SWF is spoken word frequency; HD is homophone density; PND is phonological neighborhood density; NF is neighborhood frequency; C is clustering coefficient

# Feature Reduction Using Ensemble Approach

**Yingju Xia Cuiqin Hou Zhuoran Xu Jun Sun**

Fujitsu Research & Development Center Co.,LTD.

355Unit 3F, Gate 6, Space 8,Pacific Century Place,

No.2A Gong Ti Bei Lu, Chaoyang District, Beijing 100027

{yjxia, houcuiqin, xuzhuoran, sunjun}@cn.fujitsu.com

## Abstract

The performance of many content analysis methods heavily dependent on the features they are applied. A fundamental problem that makes the content analysis difficult is the curse of dimensionality. In this study, we propose a novel feature reduction method which adopts ensemble approach to measure the divergence between the training set and test set and use the divergence to supervise the feature reduction procedure. The proposed method uses pairwise measure to get the diversity between classifiers and selects the complementary classifiers to get the pseudo labels on test set. The pseudo labels are used to measure the divergence between training set and test set. The feature reduction algorithm merges the adjacent feature space according to the divergence, such reduce the feature number. We evaluated the proposed method on several standard datasets. Experiment results shown the efficiency of the proposed feature reduction method.

## 1 Introduction

A large number of electronic textual documentations are generated everyday on webs and the Internet. For example: e-books, e-newspapers, e-magazines, and essays in blogs. It is difficult for web administrators to manage and classify numerous electronic documentations manually (Ng et al. 1997; Combarro et al. 2005;

Gao and Chien, 2012; Robati et al., 2015). It makes the content analysis tools more and more important. A main problem is the high dimensions of features which not only increase the processing time but also decrease the performance of analysis tools. Automatic feature reduction or selection methods are usually used to reduce the number of features (Reif and Shafait 2014). Removing irrelevant or redundant features not only improves performance, but also reduces the dimensionality of the data thereby shortening the training and application time of the learning scheme, building better generalizable models, and decreasing required storage. Furthermore, shorter feature vectors help the content analysis tools in better coping with the curse of dimensionality.

There is a vast literature on the feature reduction (How and Kiong, 2005; Garcia et al., 2013; Choudhary and Saraswat, 2014). When dealing with the features with continuous (real) values, the feature reduction can be regarded as discretization procedure which aim at finding a representation of each feature that contains enough information for the learning task at hand, while ignoring minor fluctuations that maybe irrelevant for that task (Ferreira and Figueiredo, 2012). In practice, discretization can be viewed as a feature reduction method since it maps data from a huge spectrum of numeric values to a greatly reduced subset of discrete values (Garcia et al., 2013).

Actually, the techniques in Garcia et al.(2013) can also be adopted to discrete values. The feature reduction task can be defined as following:

Assuming a data set consisting of  $N$  examples and  $C$  target classes, for a feature  $A$  in this data set with

continuous values which has the range  $[d_0, d_m]$ , or a set of discrete values  $(d_0, d_1, \dots, d_m)$ . The feature reduction algorithms aim to put these values into several bins or intervals:  $D = \{[d_0, d_1], [d_1, d_2], \dots, [d_{m-1}, d_m]\}$ . Each feature value is then mapped into the bin or interval in which it falls. By tuning the number of the bins, the feature space can be reduced.

Two major categories of feature reduction techniques include unsupervised and supervised methods. Unsupervised methods (Bay, 2001; Li and Wang, 2002; Yang and Webb, 2009) do not consider the class label whereas supervised ones do. (Wu, 1996; Kerber, 1992; Zighed et al., 1998; Singh and Minz, 2007; Jin et al., 2009; Jiang et al., 2010) Comprehensive listings of these techniques can be found in the works of Garcia et al. (2013). The main drawback of all the previous work is the difficulty to accurately handle the gap between the training set and test set. Once the test set changes, the previous trained model cannot catch the property of the new test set.

In this study, we propose a novel feature reduction method which adopts ensemble approach to evaluate the difference/divergence between training set and test set. The divergence is used to merge and modify the feature space, such reduce the feature number. The remaining sections of the paper are organized as follows. Section 2 presents our methods for feature reduction. Section 3 reports experimental results on standard datasets. Section 4 presents concluding remarks and future work.

## 2 Method

### 2.1 Related work

As shown by Dougherty et al. (1995), the unsupervised methods and supervised methods are different in the way they use the instance labels. The unsupervised methods do not make use of the instance labels. In contrast, supervised methods utilize the class labels of instances. The representative unsupervised method are Equal Width and Equal Frequency. The Equal Width method divides the range of observed values for a feature into  $k$  equal sized bins, where  $k$  is a user-supplied parameter. Equal Frequency method divides a continuous variable into  $k$  bins where (given  $m$  instances) each bin contains  $m/k$  (possibly duplicated) adjacent values. Take a feature which is observed to have values bounded by  $d_0$  and  $d_m$  ( $[d_0,$

$d_m]$ ), the Equal Width method computes the bin width:

$$\delta = \frac{d_m - d_0}{k}$$

The bin boundaries are constructed at  $d_0+i\delta$ , where  $i = 1, \dots, k-1$ , thus the intervals will be  $\{[d_0, d_0+\delta], (d_0+\delta, d_0+2\delta], \dots, (d_0+(k-1)\delta, d_0+k\delta]\}$

The method is applied to each feature independently. It makes no use of instance class information. Since these unsupervised methods do not utilize instance labels in setting partition boundaries, it is likely that classification information will be lost by binning as a result of combing values that are strongly associated with different classes into the same bin (Kerber, 1992). In some cases this could make effective classification much more difficult.

As mentioned above, the supervised methods utilize the instances labels to adjust the bin/interval borders. The simplest way may be to place interval borders between each adjacent pair of examples that are not classified into the same class. Suppose the pair of adjacent values on feature  $A$  are  $x_1$  and  $x_2$ ,  $x=(x_1+x_2)/2$  can be taken as an interval border. If the feature  $A$  is very informative, which means that positive and negative examples take different value intervals on the attribute, this method is very efficient and useful. However, this method tends to produce too many intervals on those attributes which are not very informative. Such many other supervised methods have been proposed. The representative method is Bayesian method (Wu, 1996).

According to Bayes formula,

$$P(c_j|x) = \frac{P(x|c_j)P(c_j)}{\sum_{k=1}^m P(x|c_k)P(c_k)} \quad (1)$$

Where  $P(c_j|x)$  is the probability of an example belonging to class  $c_j$  if the example takes value  $x$ .  $P(x|c_j)$  is the probability of the example taking value  $x$  on the feature if it is classified in the class  $c_j$ .

Given  $P(c_j)$  and  $P(c_j|x)$ , we can construct a probability curve for each class  $c_j$ :

$$B_j(x) = P(x|c_j)P(c_j) \quad (2)$$

When the curves for every class have been constructed, interval/bin borders are placed on each of those points where the leading curves are different on its two sides. Between each pair of those points including the two open ends, the learning curve is the same.

## 2.2 Motivation

From the description in Section 2.1, we know that the supervised methods consider the class attribute depends on the interaction between input features and class labels. It depends on the stationary assumption. Actually, the stationary assumption does not always hold in the real applications (Bai et al., 2014; Gama et al. 2014). For many learning tasks where data is collected over an extended period of time, its underlying distribution is likely to change. The drift in the underling distribution may result in a change in the learning problem.

If we can get the real labels in the test set, we should utilize these labels to supervise the feature reduction. But actually, we can't get the real labels. Consider that there is always a pool of classifiers such as Random Forest, Gradient Boosting, Maximum Entropy and Naïve Bayes. Each classifier has its own advantage. The ensemble learning (Dietterich, 2000; Wozniak et al., 2014) is such a technique focus on the combination of classifiers from heterogeneous or homogeneous modeling background to give the final decision. It is primarily used to improve the classification performance of a model, or reduce the likelihood of an unfortunate selection of a poor one. Dietterich (2000a) summarized the benefits:

(a) Allowing to filter out hypothesis that, though accurate, might be incorrect due to a small training set.

(b) Combining classifiers trained starting from different initial conditions could overcome the local optima problem.

(c) The true function may be impossible to be modeled by any single hypothesis, but combinations of hypotheses may expand the space of representable functions.

In this study, we adopt the ensemble learning method to the feature reduction. We employ ensemble classifiers to process the test set and get classification labels. We call the labels gotten from this procedure the pseudo labels since they are not the real labels in the test set. The pseudo labels are utilized to measure the difference/divergence between the training set and test set. The difference is been used to modify the feature space. More concretely, for each adjacent interval, the proposed method calculates the divergence between the labeled examples in training set and the pseudo

labeled examples in test set and decide whether merge these intervals or not.

## 2.3 Method

To simplify, we take the two-class classification as example. The task of feature reduction is to put the feature values into several bins. The feature number will be reduced since the number of bins is generally less than the feature value number.

The typical unsupervised method such as the Equal Width method, do not make use of instance labels. The feature values are put into several equal sized bins. The supervised methods try to utilize the distribution of the classes in the training set to supervise the feature merge procedure. The equal-width method has the risk that merges values that are strongly associated with different classes into the same bin. The representative supervised method such as Bayesian avoids this problem by estimating the condition probability in the training set. The basic assumption is that the training set and test set has the same distribution, but it does not always holds. When distribution of training and test set are difference, the typical supervised method will fail.

The ensemble learning approach is adopted in this study, we get the pseudo labels of every instance in the test set by using other classifiers. Then, we use the *KL* divergence to measure the difference between training set and test set.

$$D(P_{tr} \parallel P_{ts}) = \sum_y \sum_i P_{tr}(y|f_i) \log \frac{P_{tr}(y|f_i)}{P_{ts}(y|f_i)} \quad (3)$$

Here the  $f_i$  denote the feature  $i$ , the  $P_{tr}(y|f_i)$  and  $P_{ts}(y|f_i)$  are the probability of the output label under the condition  $f_i$  in the training set and test set respectively, the  $D(P_{tr} \parallel P_{ts})$  is the divergence between the training set and test set in the given interval.

Since the pseudo labels are the crucial to the feature reduction, how to select the candidate classifiers for getting the pseudo labels is also the key point. The intuition is that the mutually complementary classifiers which are characterized by high diversity and accuracy should be selected to get the pseudo labels for each other. Actually, the diversity has been recognized as a very important characteristic in classifier combination. Empirical results have illustrated that there exists positive correlation between accuracy of the ensemble and diversity among the base cassifiers (Dietterich,

2000b; Kuncheva and Whitaker, 2003; Tang *et al.*, 2006). Further, most of the existing ensemble learning algorithms (Brieman, 1996; Liu *et al.* 2000) can be interpreted as building diverse base classifiers implicitly. However, the problem of measuring classifier diversity and so using it effectively for building better classifier ensembles is still an open topic. Most researchers discuss the concept of diversity in terms of correct/incorrect outputs (Brown *et al.*, 2005; Kuncheva and Whitaker, 2003; Tang *et al.*, 2006). Kuncheva and Whitaker (2003) divide the diversity measures into pairwise diversity measures and non-pairwise diversity measures. For pairwise diversity measure, the  $Q$  statistics, the correlation coefficient, the disagreement measure and the double-fault measure are most commonly used. The previous experimental studies have shown that most diversity measures perform similarly (Kuncheva and Whitaker, 2003; Tang *et al.*, 2006). In this study, we adopt the disagreement measure (Ho, 1998; Skalak, 1996) to select the classifiers for getting pseudo labels.

The disagreement measure of classifier  $i$  and  $k$  is defined as :

$$Dis_{i,k} = \frac{N^{01} + N^{10}}{N^{00} + N^{01} + N^{10} + N^{11}} \quad (4)$$

Where  $N^{00}$ ,  $N^{01}$ ,  $N^{10}$  and  $N^{11}$  are derived from the below table:

	$D_k$ correct(1)	$D_k$ wrong(0)
$D_i$ correct(1)	$N^{11}$	$N^{10}$
$D_i$ wrong(0)	$N^{01}$	$N^{00}$

Table 1: A 2\*2 table of the relationship between a pair of classifiers

Support we have gotten the  $L$  classifiers which have high diversity with the target classifier for feature space reduction. The straightforward way is to use the classifier with highest diversity to get the pseudo labels. However, this method does not consider the accuracy of the classifier been selected. How about the result if the classifier with the highest diversity does not performance well? Actually, beside the diversity, the accuracy of the classifier and the classification confidence are also key factors for the pseudo labels getting. The accuracy of classifier can be explicitly expressed by the weight of classifier. The classification

confidence, which was theoretically proved to be a key factor on the generalization performance (Shawe-Taylor and Cristianini, 1999), has been utilized in certain ensemble learning algorithms (Freund and Schapire, 1997; Li *et al.*, 2014; Quinlan, 1996; Schapire and Singer, 1999).

In this study, we extract the pseudo labels by combining the ensemble margin (Schapire *et al.*, 1998) and classification confidence (Li *et al.*, 2014).

Let:

$h_j$  ( $j=1,2, \dots, L$ ): the selected classifiers with high diversity.

$X=\{(x_i, y_i), i=1,2, \dots, n\}$ : the data set

$y_i$ : the class label of the sample  $x_i$

$\bar{y}_{ij}$ : the classification decision of  $x_i$  estimated by the classifier  $h_j$

$c_{ij}$ : the classification confidence of  $x_i$  estimated by the classifier  $h_j$

define the margin as:

$$m(x_i) = \sum_{j=1}^L w_j \gamma_{ij} c_{ij} \quad \text{s.t. } w_j \geq 0, \quad \sum_{j=1}^L w_j = 1 \quad (5)$$

where the  $w_j$  is the weight of the classifier  $h_j$  and

$$\gamma_{ij} = \begin{cases} 1 & \text{if } y_i = \bar{y}_{ij} \\ -1 & \text{if } y_i \neq \bar{y}_{ij} \end{cases} \quad (6)$$

We can get the optimal  $W = [w_1, \dots, w_L]^T_{L*1}$  by minimizing the objective function below:

$$W = \underset{W}{\operatorname{argmin}} \|U - TW\|_2^2 + \lambda \|W\|_2 \quad (7)$$

Where  $U = [1, \dots, 1]^T_{n*1}$ ,  $T = [\gamma_{ij} c_{ij}]_{n*L}$

$$\|U - TW\|_2^2 = \sum_{i=1}^n (1 - m(x_i))^2 \quad (8)$$

$\lambda$  is a Lagrange multiplier

The optimal  $W$  is utilized to get the final pseudo labels by combine the  $L$  classifiers with high diversity.

Once we got the pseudo labels, we will use these labels to supervise the feature reduction procedure. The distribution difference between the training set and test set can be measured.

The proposed feature reduction method searches the whole feature space by a fixed step. For each adjacent interval, the proposed method calculates the divergence between the labeled examples in training set and the pseudo labeled examples in test set and decides whether merge these intervals or not.

The adjacent intervals which have small change in the distribution will be merged. By elaborately selected moving step and the distribution distance threshold, the feature space will finally partitioned into several sub-space which will reduce the original feature space.

The algorithm is shown below:

**BEGIN**

For each classifier  $i$ :

    Select the  $L$  classifiers with high disagreement with the classifier  $i$  in the classifier pool

    Optimize the weight  $W$  of the selected  $L$  classifiers

    Make an ensemble model form the selected  $L$  classifiers and the optimal weight  $W$

    Get the pseudo labels in the test set using the ensemble model

For each feature  $f_i$ :

    Set the interval merge step:  $T$

    For each adjacent  $T$ :

        Get the Bayesian measure  $B_T$  using the formula (2)

        Get  $KL$  Divergence  $D_p$  using the formula (3)

        IF  $B_T < \theta_b$  and  $D_p < \theta_d$

            Merge the adjacent intervals

        ELSE

            Go to next interval  $T$

**END**

Here, the  $\theta_b$  and  $\theta_d$  are the threshold for Bayesian-measure and  $KL$  divergence respectively.

### 3 Experimental Results

The performance of the proposed method is evaluated on 20 UCI datasets (Frank and Asuncion, 2010). The detailed information of these datasets are shown in Table 2.

In the table 2, '#I' denotes the number of instances, '#F' denotes the feature number and '#C' denotes the class number. These datasets cover some high-dimensional sets, some large sets, some small sets and some typical/balanced sets. More detailed information can be found on the UCI website.

The classifier pool includes Random Forest, Decision Tree, Gradient boosting, Maximum Entropy and Naïve Bayes. Every model uses the pseudo labels gotten from others to make the feature reduction.

A set of experiments are conducted in the multiple classifier system to show the performance

of the proposed ensemble feature reduction method. The conventional weighted majority voting approach is adopted as the fusion method for multiply classifier. Some analysis (Kuncheva, 2004; Wozniak and Jackowski, 2009) shown that it is an effective way for fusion of multiply classifier. The algorithm begins by creating a set of experts and assigning a weight to each. When a new instance arrives, the algorithm passes it to and receives a prediction from each expert. The algorithm predicts based on a weighted majority vote of the expert predictions.

The data sets considered are partitioned using the 10-fold cross-validation procedure. The 'Accuracy' is used as the performance measures. The 'Accuracy' is the number of successful hits relative to the total number of classification. It has been by far the most commonly used metric for assessing the performance of classifiers for years (Prati et al., 2011; Witten et al., 2011).

Dataset	#I	#F	#C
Abalone	4177	8	28
Audiology	226	69	23
Breast Cancer	286	9	2
Car Evaluation	1728	6	4
Census	199523	40	2
Ecoli	336	8	8
Internet Advertisements	3279	1558	2
Iris	150	4	3
Letter Recognition	20000	16	26
Magic Gamma Telescope	19020	11	2
Mammographic Mass	961	6	2
Molecular Biology	3190	61	3
Musk	476	168	2
Nursery	12960	8	5
Ozone Level Detection	2536	73	2
Page Blocks Classification	5473	10	5
Pima Indians Diabetes	768	8	2
Spectf Heart	267	44	2
Statlog (Vehicle Silhouettes)	946	18	4
Yeast	1484	8	10

Table 2. The datasets description

The experimental results on very data set are shown on Table 3. Here, the proposed ensemble method is compared with the typical unsupervised method EW (Equal Width) and the typical supervised method Bayes (Bayesian). The experimental results show that the proposed ensemble method outperform the conventional method (Equal Width and Bayesian) on almost all data set except the 'Iris' data set.

By analysis of the size of dataset, we found that the dataset size will impact the performance. Take the 'Iris' as example, there are only 150 instances in this dataset which lead to a small feature space (only 22 unique values for the first feature). There is little hint to make the feature reduction. It is very difficult to put them into several bins.

Dataset	EW	Bayes	Ensemble
Abalone	87.86	88.62	<b>89.58</b>
Audiolog	59.13	59.6	<b>60.06</b>
Breast Cancer	90.6	91.65	<b>92.21</b>
Car Evaluation	84.24	85.12	<b>86.19</b>
Census	84.04	84.39	<b>86.53</b>
Ecoli	77.5	78.35	<b>78.89</b>
Internet	64.09	64.64	<b>65.85</b>
Iris	<b>95.5</b>	94.25	94.25
Letter	87.59	88.21	<b>90.26</b>
Magic	86.72	87.82	<b>90.09</b>
Mammographic	67.6	68.05	<b>69.01</b>
Molecular	70.89	71.64	<b>72.83</b>
Musk	84.42	84.61	<b>85.35</b>
Nursery	83.59	84.29	<b>86.05</b>
Ozone	73.02	73.26	<b>74.95</b>
Page	83.72	84.16	<b>85.63</b>
Pima	69.07	69.52	<b>70.61</b>
Spectf Heart	80.57	80.72	<b>81.19</b>
Statlog	89.05	89.35	<b>90.61</b>
Yeast	60.99	61.84	<b>62.65</b>

Table 3. The experimental results

Since the feature space reduction is conducted on the feature space for each classifier. To further investigate the performance of the proposed feature

reduction method, the compared experiments on each single classifier are also conducted to show the effect of the proposed method. Here, we take the Equal Width as the baseline method and the relative difference is taken as the evaluation measure.

The relative difference is calculated as:

$$\frac{Accuracy_{ref} - Accuracy_{baseline}}{Accuracy_{baseline}} \tag{9}$$

Here, the  $Accuracy_{baseline}$  is the accuracy of EW on each dataset. The  $Accuracy_{ref}$  is the accuracy of Bayesian and the proposed ensemble method.

Figure 1 ~ 6 show the experimental results on each individual classifier (Random Forest, Decision Tree, Gradient boosting, Maximum Entropy and Naïve Bayes). Here, the baseline method is Equal Width. The blue line is the relative difference of Bayesian method comparing with the baseline. The red line is the relative difference of the Ensemble method. The x-axis shows the name of the selected datasets which are sorted by the size. The smallest dataset is 'Iris' which only has 150 instances while the largest dataset is the 'Census' dataset which has 199,523 instances.

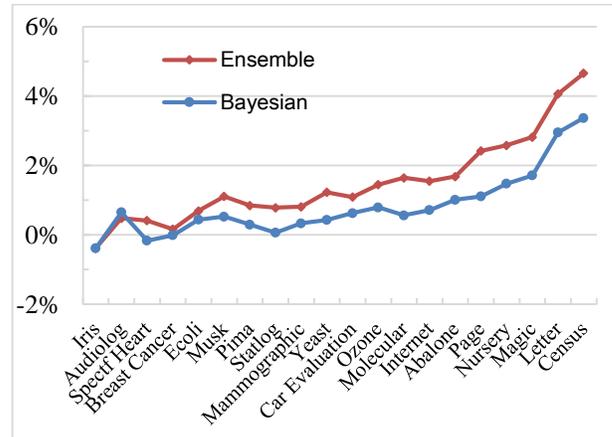


Figure 1. The experimental results on Random Forest

From Figure 1, we see that for Random Forest classifiers, the more data, the better performance. More than 4% enhancement has been achieved on the 'Census' dataset which has 199,523 instances. In most dataset, the proposed ensemble method and Bayesian method are better than the unpervised method Equal Width. When the dataset is small, the performance is not so satisfied. For example, the ensemble method and Bayesian method worse than

the Equal Width method on the 'Iris' dataset. Also we can see that, when the dataset is small, the ensemble method can not beat the Bayesian method ('Audiolog': 226 instances).

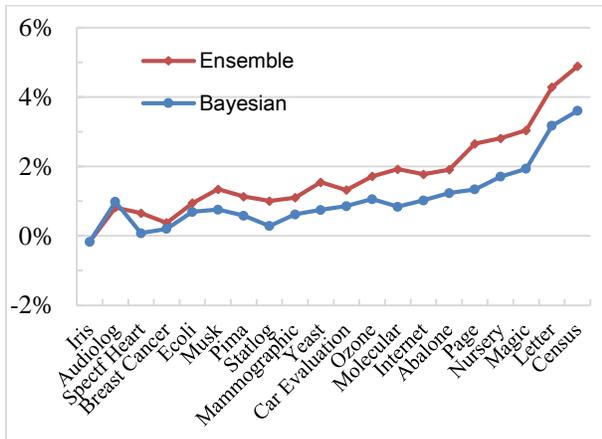


Figure 2. The experimental results on Decision Tree

Figure 2 shows the experimental results using Decision Tree classifier. We can see that the same trend as shown on the Random Forest. The highest enhancement is about 5% which is a little high than Random Forest. It is also gotten from the 'Census' dataset.

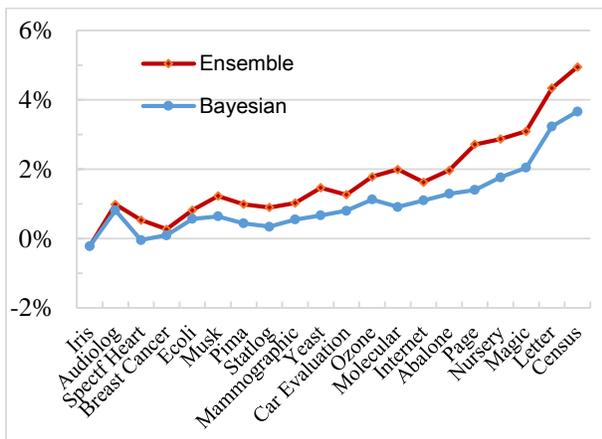


Figure 3. The experimental results on Gradient Boosting

Figure 3 shows the experimental results using Gradient Boosting classifiers. It's similar with the Random Forest and Decision Tree. For Gradient Boosting classifier, the ensemble method also does not performance well on the small datasets.

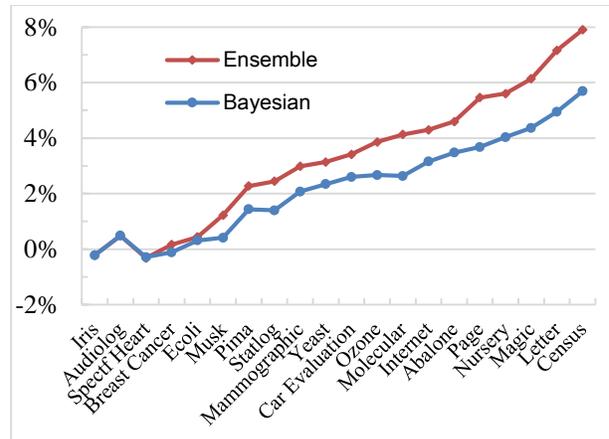


Figure 4. The experimental results on Maximum Entropy

Figure 4 shows the experimental results using Maximum Entropy classifier. The proposed ensemble method achieved about 8% enhancement when the dataset is large ('Census': 199,523 instances). However, the performance also fluctuates when the dataset is small. It becomes stable when the dataset size is larger than 500. This may be because the ensemble method needs more data to measure the distribution divergence between training set and test set.

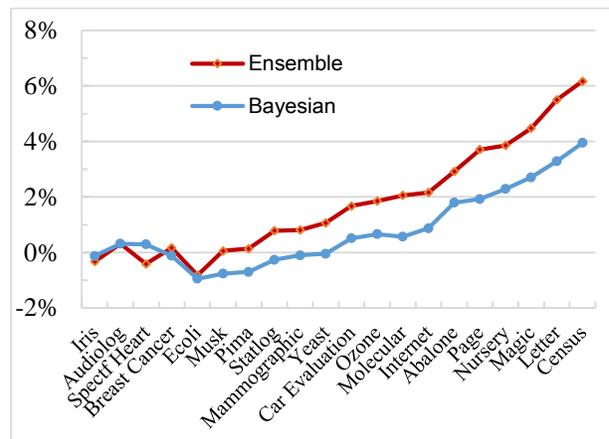


Figure 5. The experimental results on Naïve Bayes

Figure 5 shows the experimental results using Naïve Bayes classifier. The enhancement is also great (more than 6%). It is more fluctuant than the Maximum Entropy classifier when the dataset is small.

To further investigate the performance on different data size. A set of experiments on 'Census'

dataset are conducted. The sub-datasets range from 50 to 190,000 are extracted from the whole dataset. The experiments are intend to compare the performance of EW, Bayesian and the proposed ensemble method. The experimental results are shown as the relative difference with the baseline method (Equal Width method).

Figure 6 shows the experimental results. The x-axis shows the size of each sub-datasets. The y-axis shows the relative difference. The experiments are conducted in the multiply classifier scenario, that is the finnal predcition is made by the ensemble classifier. We can see that the total enhancement is not higher than the Maximum Entropy or the Naïve Bayes classifier. This is because the fusion procedure highly depends on the diversity among the classifiers. It can't get the highest enhancement as the single classifier.

When the data size is small, both the proposed ensemble and Bayesian method cannot get good performance. For example, when the data size is less than 100, the ensemble and Bayesian methods are worse than EW. It is because that the Bayesian method needs to make statistic on the training set. The ensemble method need more data to calculate the distribution difference between training set and test set. From the Figure 6, we can see that, even there are about 1,000 samples, the ensemble method cannot get great enhancement in comparison with the Bayesian method. The ensemble method is worse than Bayesian method when the data size is small than 200. With the bigger dataset, the ensemble method performance better, about 4% enhancement can be achieved.

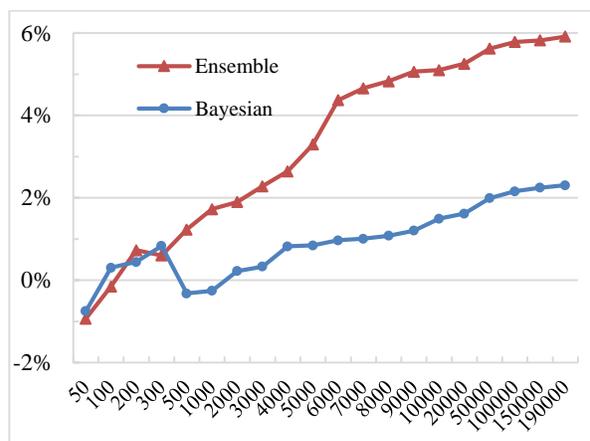


Figure 6. The experimental results on dataset size

## 4 Conclusions and Future Work

In this study, we propose a feature reduction method which uses ensemble approach to get the pseudo labels and utilize the pseudo labels to supervise the feature reduction procedure. The experiments conducted on different type of datasets compared the proposed method with the conventional feature reduction methods. The experimental results shown the effectiveness and efficiency of the proposed method.

The future work includes the scheme on selecting the candidate models for getting the pseudo labels. The measurement on distribution difference between training set and test set also need to be explored. How to improve the performance on small datasets is also research topic.

## References

- Bai Q. X., Lam H. and Sclaroff S. 2014. A Bayesian Framework for Online Classifier Ensemble. The 31st International Conference on Machine Learning, pages 1584-1592, Beijing, China, 2014.
- Bay S. D. 2001. Multivariate Discretization for set Mining. Knowledge information Systems, Vol. 3, pp 491-512
- Breiman L. 1996. Bagging predictors. Machine Learning, 24(2), 1996, 123-140
- Brown G., Wyatt J., Harris R. and Yao X. 2005. Diversity creation methods: a survey and categorization. Journal of Information Fusion 6(1), 2005, 5-20.
- Choudhary A., and Saraswat J. K. 2014. Survey on Hybrid Approach for Feature Selection. International Journal of Science and Research, 3(4), 438-439.
- Combarro E. F., Montan E., D' Iaz I., Ranilla J., and Mones R. 2005. Introducing a Family of Linear Measures for Feature Selection in Text Categorization. IEEE Transactions on Knowledge and Data Engineering, Vol. 17, No. 9, pp. 1223-1232
- Dietterich T. 2000a. Ensemble methods in machine learning, in: Multiple Classifier Systems. Lecture Notes in Computer Science, vol. 1857, Springer, Berlin, Heidelberg, 2000, 1-15.
- Dietterich T. 2000b. An experimental comparison of three methods for constructing ensembles of decision trees: Bagging, boosting and randomization. Machine Learning, 40(1), 2000, 1-22.
- Dougherty J., Kohavi R., and Sahami M. 1995. Supervised and unsupervised discretization of

- continuous features. In *Machine learning: proceedings of the twelfth international conference*, Vol. 12, pp 194-202
- Ferreira A. J. and Figueiredo M. A. T. 2012. An unsupervised approach to feature discretization and selection. *Pattern Recognition* 45(2012), pp. 3048–3060
- Frank A. and Asuncion A. 2010. UCI machine learning repository, <http://archive.ics.uci.edu/ml>.
- Freund Y. and Schapire R. E. 1997. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of computer and system sciences*, 55(1), 1997, 119-139.
- Gama J., zliobaite I., Bifet A., Pechenizkiy M. and Bouchachia A. 2014. A survey on concept drift adaptation. *ACM Computing Surveys* 46.4 (2014): 44.
- Gao L. J. and Chien B. C. 2012. Feature Reduction for Text Categorization Using Cluster-Based Discriminant Coefficient. In *Technologies and Applications of Artificial Intelligence (TAAI), 2012 Conference on* (pp. 137-142). IEEE.
- Garcia S., Luengo J., Sez J., Lpez V. and Herrera F. 2013. A survey of discretization techniques: Taxonomy and empirical analysis in supervised learning. *IEEE Transactions on Knowledge and Data Engineering* 25, pp. 734–750.
- Ho T. 1998. The random space method for constructing decision forests. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20:8, 1998, 832–844.
- How B. C. and Kiong W. T. 2005. An examination of feature selection frameworks in text categorization. In *AIRS'05: Proceedings of 2nd Asia information retrieval symposium*, PP 558–564.
- Kerber R. 1992. ChiMerge: Discretization of Numeric Attributes. *Proc. Nat'l Conf. Artificial Intelligence Am. Assoc. for Artificial intelligence*, pp 123-128
- Kuncheva L. 2004. *Combining Pattern Classifiers: Method and Algorithms*, Wiley Interscience, 2004
- Kuncheva L. and Whitaker C. 2003. Measures of diversity in classifier ensembles and their relationship with the ensemble accuracy. *Machine Learning*, 51:181-207, 2003.
- Jiang F., Zhao Z., and Ge Y. 2010. A Supervised and Multivariate Discretization Algorithm for Rough Sets. *Proc. Fifth Int'l Conf. Rough Set and Knowledge Technology (RSKT)*, pp. 596-603
- Li L., Hu Q., Wu X. and Yu D. 2014. Exploration of classification confidence in ensemble learning. *Pattern Recognition*, 47(9), 2014, 3120-3131.
- Li R. P. and Wang Z. O. 2002. An Entropy-based Discretization Method for Classification Rules with Inconsistency Checking. *Proc. First Int'l Conf. Machine Learning and Cybernetics*. pp 243-246
- Liu H. and Setiono R. 1997. Feature selection via discretization. *IEEE transactions on knowledge and data engineering*, Vol 9, No. 4, 642-645
- Liu Y. Yao X. and Higuchi T. 2000. Evolutionary ensembles with negative correlation learning. *IEEE Transactions on Evolutionary Computation*, 4, 2000, 380-387
- Jin R., Breitbart Y. and Muoh C. 2009. Data Discretization Unification. *Knowledge and Information Systems*, Vol. 19, pp 1-29
- Ng H. T., Goh W. B., and Low K. L. 1997. Feature selection, perceptron learning, and a usability case study for text categorization. In *SIGIR '97: Proceedings of the 20th annual international ACM SIGIR conference on Research and development in information retrieval*, PP 67–73
- Prati R.C., Batista G.E.A.P.A., and Monard M.C. 2011. A Survey on Graphical Methods for Classification Predictive Performance Evaluation, *IEEE Trans. Knowledge and Data Eng.*, Vol. 23, No. 11, pp. 1601-1618, Nov. 2011, doi: 10.1109/TKDE.2011.59.
- Quinlan J. R. 1996. Bagging, boosting, and C4. 5. In *AAAI/IAAI*, Vol. 1, 1996, 725-730
- Reif M. and Shafait F. 2014. Efficient feature size reduction via predictive forward selection, *Pattern Recognition(2014)*, Vol. 47, PP 1664-1673
- Robati, Z., Zahedi, M., and Fayazi Far, N. 2015. Feature Selection and Reduction for Persian Text Classification. *International Journal of Computer Applications*, 109(17), 1-5.
- Schapire R. E., Freund Y., Bartlett P. and Lee W. S. 1998. Boosting the margin: A new explanation for the effectiveness of voting methods. *Annals of statistics*, 1998, 1651-1686.
- Schapire R. E. and Singer Y. 1999. Improved boosting algorithms using confidence-rated predictions. *Machine learning*, 37(3), 1999, 297-336.
- Shawe-Taylor J. and Cristianini N. 1999. Robust bounds on generalization from the margin distribution. *The 4th European Conference on Computational Learning Theory*, 1999.

- Singh G. K. and Minz S. 2007. Discretization Using Clustering and Rough Set Theory. Proc. 17th int'l Conf. Computer Theory and Applications, pp 330-336
- Skalak D. 1996. The sources of increased accuracy for two proposed boosting algorithms. In Proc. American Association for Artificial Intelligence, AAAI-96, Integrating Multiple Learned Models Workshop
- Tang E. K., Suganthan P. N. and Yao X. 2006. An analysis of diversity measures. Mach. Learn. 65(2006)247–271.
- Witten I.H., Frank E., and Hall M.A. 2011. Data Mining: Practical Machine Learning Tools and Techniques, third ed. Morgan Kaufmann, 2011.
- Wozniak M., Grana M. and Corchado E. 2014. A survey of multiple classifier systems as hybrid systems. Information Fusion, 16:3-17, 2014.
- Wozniak M. and Jackowski K. 2009. Some remarks on chosen methods of classifier fusion based on weighted voting, in: E. Corchado, X. Wu, E. Oja, A. Herrero, B. Baruque (Eds.), Hybrid Artificial Intelligence Systems, Lecture Notes in Computer Science, vol. 5572, Springer, Berlin/Heidelberg, 2009, pp. 541–548
- Wu X. 1996. A Bayesian discretizer for real-valued attributes. The Computer Journal, 39(8), 688-691.
- Yang Y. and Webb G. I. 2009. Discretization for Naive-Bayes Learning: Managing Discretization bias and Variance. Machine Learning. vol. 74, No. 1, pp 39-74
- Zighed D. A., Rabaseda S. and Rakotomalala R. 1998. FUSINTER: A method for discretization of continuous Attributes. Int'l J. Uncertainty, Fuzziness Knowledge-based Systems, Vol. 6, pp 307-326

# Measuring Popularity of Machine-Generated Sentences Using Term Count, Document Frequency, and Dependency Language Model

Jong Myoung Kim<sup>1</sup>, Hancheol Park<sup>2</sup>, Young-Seob Jeong<sup>1</sup>

Ho-Jin Choi<sup>1</sup>, Gahgene Gweon<sup>2,3</sup>, and Jeong Hur<sup>3</sup>

<sup>1</sup>School of Computing, KAIST

<sup>2</sup>Department of Knowledge Service Engineering, KAIST

<sup>3</sup>Knowledge Mining Research Team, ETRI

{grayapple, hancheol.park, pinode, hojinc, ggweon}@kaist.ac.kr  
jeonghur@etri.re.kr

## Abstract

We investigated the notion of “popularity” for machine-generated sentences. We defined a popular sentence as one that contains words that are frequently used, appear in many documents, and contain frequent dependencies. We measured the popularity of sentences based on three components: *content morpheme count*, *document frequency*, and *dependency relationships*. To consider the characteristics of agglutinative language, we used content morpheme frequency instead of term frequency. The key component in our method is that we use the product of content morpheme count and document frequency to measure word popularity, and apply language models based on dependency relationships to consider popularity from the context of words. We verify that our method accurately reflects popularity by using Pearson correlations. Human evaluation shows that our method has a high correlation with human judgments.

## 1 Introduction

Natural language generation is widely used in variety of Natural Language Processing (NLP) applications. These include paraphrasing, question answering systems, and Machine Translation (MT). To improve the quality of generated sentences, arranging effective evaluation criteria is critical (Callison-Burch et al., 2007).

Numerous previous studies have aimed to evaluate the quality of sentences. The most frequently used evaluation technique is asking judges to score

those sentences. Unlike computer algorithms, humans can notice very delicate differences and perceive various characteristics in natural language sentences. Conventional wisdom holds that human judgments represent the gold standard; however, they are prohibitively expensive and time-consuming to obtain.

Because of the high cost of manual evaluation, automatic evaluation techniques are increasingly used. These include very popular techniques that measure meaning adequacy and lexical similarity, such as BLEU (Papineni et al., 2002), METEOR (Banerjee and Lavie, 2005), and TER plus (Snover et al., 2009). Additionally, a distinctive characteristic of auto evaluation techniques is that they can be applied not only to performance verification, but also to the generation stage of NLP applications. Although these techniques can make experiments easier and accelerate progress in a research area, they employ fewer evaluation criteria than humans.

In general, previous research efforts have focused on “technical qualities” such as meaning and grammar. However, customer satisfaction is sometimes determined more by “functional quality” (how the service work was delivered) than by “technical quality” (the quality of the work performed) (Mittal and Lassar, 1998). Especially, Casaló et al. (2008) showed that the customers’ loyalty and satisfaction are affected by their past frequent experiences. We focused on this aspect and propose a new criterion, popularity, to consider the functional quality of sentences. We define a popular sentence as one that contains words that are frequently used, appear in many documents, and contain frequent dependencies. Us-

ing this definition, we aim to measure the popularity of sentences.

In this paper, we investigate the notion of “*popularity*” for machine-generated sentences. We measured popularity of sentences with an automatic method that can be applied to the generation stage of MT or paraphrasing. Because it is a subjective evaluation, measuring the popularity of sentences is a difficult task. We defined a popular sentence as one that contains words that are frequently used, appear in many documents, and contain frequent dependencies. Subsequently, we began our analysis by calculating Term Frequency (TF). To reflect the characteristics of agglutinative languages, we apply a morpheme analysis during language resources generation. As a result, we obtain a Content Morpheme Count (CMC). To complement areas CMC cannot cover (words that have abnormally high CMC), we apply morpheme-based Document Frequency (DF). Lastly, to consider popularity came from contextual information, we apply a dependency relationship language model. We verify our method by analyzing Pearson correlations between human judgments; human evaluation shows that our method has a high correlation with human judgments. And our method shows the potential for measuring popularity by involving the contextual information.

The remainder of this paper is organized as follows. Section 2 presents related works in the field of sentence evaluation. Section 3 explains the approach to measure the popularity of words and sentences. In Section 4, we evaluate the usefulness of our method. In section 5, we analyze the result of experiment Lastly, Section 6 concludes the paper.

## 2 Related Works

Manual evaluation, the most frequently used technique, asks judges to score the quality of sentences. It exhibits effective performance, despite its inherent simplicity. Callison-Burch asked judges to score fluency and adequacy with a 5-point Likert scale (Callison-Burch et al., 2007), and asked judges to score meaning and grammar in a subsequent paper (Callison-Burch, 2008). Similarly, Barzilay et al. asked judges to read hand-crafted and application-crafted paraphrases with corresponding meanings, and to identify which version was most readable and

best represented the original meaning (Barzilay and Lee, 2002). Philip M. Mc et al. studied overall quality using four criteria (McCarthy et al., 2009). Using these evaluation techniques, humans can identify characteristics that machines cannot recognize, such as nuances and sarcasm. Overwhelmingly, humans are more sensitive than computers in the area of linguistics. As a result, manual evaluation provides the gold standard. However, manual evaluation presents significant problems. It is prohibitively expensive and time-consuming to obtain.

To address these limitations, there have been studies involving automatic evaluation methods. Papieni et al. (2002) and Callison-Burch et al. (2008) proposed methods that measure meaning adequacy based on an established standard. Several methods based on Levenshtein distance (Levenshtein, 1966) calculate superficial similarity by counting the number of edits required to make two sentences identical (Wagner and Fischer, 1974; Snover et al., 2009). These methods can be used to calculate dissimilarity in paraphrasing. Chen et al. measured paraphrase changes with n-gram (Chen and Dolan, 2011). These automatic evaluations also present a problem — the absence of diversity. There are many senses humans can detect from sentences, even if they are not primary factors such as meaning adequacy or grammar. We identify a novel criteria, popularity, as one of those senses, based on the fact that customer satisfaction is sometimes derived from functional quality (Mittal and Lassar, 1998).

We define the popularity of a sentence using TF, DF and dependency relations. TF, defined as the number of times a term appears, is primarily used to measure a term’s significance, especially in information retrieval and text summarization. Since Luhn used total TF as a popularity metric (Luhn, 1957), TF has been frequently used to measure term weight, and employed in various forms to suit specific purposes. Term Frequency-Inversed Document Frequency (TF-IDF), the most well-known variation of TF, is used to identify the most representative term in a document (Salton and Buckley, 1988). Most previous research using those variations has focused on the most significant and impressive terms. There has been minimal research concerned with commonly used terms. We measured popularity of sentences with these commonly used terms that

have high TF and DF.

### 3 Method

In this section, we explain the process of language resource generation, and propose a method to measure the popularity of sentences. First, we utilize morpheme analysis on the corpus of sentences, because our target language is Korean which is an agglutinative languages. Next, we statistically analyze each content morpheme occurrence, and then calculate sentence popularity using these resources.

#### 3.1 Korean Morpheme Analysis

We built our language resources (Content Morpheme Count-Document Frequency (CMC-DF) and Dependency Language Model (DLM)) by analyzing a massive corpus of Korean sentences statistically. Because Korean is an agglutinative language, we needed to conduct morpheme analysis before we built those resources. In agglutinative language, words can be divided into content morphemes and an empty morpheme. Content morphemes contain the meaning of words, while empty morphemes are affixed to the content morpheme to determine its grammatical role. For example, in the sentence “뉴욕에 가다. (Go to New York City.)”, a word “뉴욕에” can be divided into “뉴욕” and “에”. A content morpheme “뉴욕” means “New York city” and an empty morpheme “에” do the role of a stop word “to”. Because there are numerous combinations of two morpheme types, it is not appropriate to compile statistics on the words without morpheme analysis. Via this process, we can disassemble a word into a content morpheme and empty morpheme, and obtain a statistical result that accurately represents the word. Postpositions and endings, the stop words of Korean, are filtered in this process. Additionally, we conduct conjunctions filtering, most of stop words of Korean are eliminated in morpheme analysis and filtering. We used a Korean morpheme analyzer module created by the Electronics and Telecommunications Research Institute (ETRI)<sup>1</sup>.

#### 3.2 Measuring Word Popularity

Before calculating the popularity of sentences, we attempt to measure the popularity of words. We de-

finied a popular word as one with a frequently used content morpheme. The empty morphemes are not considered, because they are stop words in Korean. We adopt Content Morpheme Count (CMC), a variation of TF, to measure usage of the content morpheme of words. CMC is the frequency of a word’s content morpheme in a set of documents. The CMC of the word  $w$  is driven in the following equations.

$$CMC_w = \max(0, \log b(w)) \quad (1)$$

$$b(w) = \sum_{d \in D} f_{m,d} \quad (2)$$

In Eq. (2),  $b(w)$  is the qualified popularity of word  $w$ , defined as the number of content morphemes  $m$  of word  $w$  in entire documents  $D$ .  $f$  is the frequency of a particular content morpheme  $m$  in document  $d$ . We applied the logarithm in Eq. (1) because simple frequency measures have a tendency to emphasize high-frequency terms. Furthermore, we utilize the max function to handle unseen morphemes in the training data corpus.

$$B(s) = \frac{\sum_{i=1}^n CMC_{w_i}}{n} \quad (3)$$

Using the average of CMC, we measure the popularity  $B(s)$  of sentences with a size of  $n$  in Eq. (3). We use this score as a baseline.

Unfortunately, CMC is not sufficient to reflect the popularity of words, because there are some cases it cannot cover. Technical terms or named entities frequently occur only in a few documents such as scientific articles or encyclopedia entries. In those cases, a high CMC is calculated, even if those words are not popular to the general public. For example, the word “스타매거진 (star magazine)” occurred 270 times in only one document (Korean Wikipedia contains 700,000 documents). Similarly, the word “글루코코르티코이드 (glucocorticoid)” occurred 39 times in only one document (the common word “카펫 (carpet)” occurred 41 times over 36 documents). The CMCs of those words are relatively high, but they are not popular terms to ordinary people.

Thus, we inversely applied the concept of TF-IDF. TF-IDF assumes that if a term frequently occurs in only a few documents, the term is significant in those documents. Inversely, we assumed that if a content

<sup>1</sup><https://www.etri.re.kr/kor/main/main.etri>

morpheme is used frequently and occurs in numerous documents, that morpheme is popular to people. We quantified the popularity of words as the number of documents in which its content morpheme occurs. We calculate Document Frequency (DF) in the following equations.

$$DF_w = \max(0, \log c(w)) \quad (4)$$

$$c(w) = |\{d \in D : m \in d\}| \quad (5)$$

In Eq. (5),  $c(w)$  is the number of documents  $d$  in which content morpheme  $m$  occurs. Similarly, logarithm and max function are applied in Eq. (4).

Using the notion of CMC and DF, we defined the popularity of words  $f(w)$  as follows in Eq. (6). Lastly, we measured the popularity of sentences by calculating the average popularity of words in sentences with a size of  $n$ . Popularity of sentences  $F(s)$  based on word popularity is represented in Eq. (7)

$$f(w) = CMC_w \times DF_w \quad (6)$$

$$F(s) = \frac{\sum_{i=1}^n f(w_i)}{n} \quad (7)$$

### 3.3 Measuring Context Popularity

We measure popularity of sentence using word popularity in the previous section. Nevertheless, there is another element as important as word popularity. The element is whether the word is suitable in the context. For example, we frequently use “powerful,” not “strong,” when discussing a computer having substantial computational ability. As an adjective, “powerful” and “strong” have similar meanings and popularity. The use of “strong” is perhaps more frequent than that of “powerful.” However, if we consider the context of a noun “computer” joined with each word, i.e., “powerful computer” and “strong computer”, there will be a significant difference in popularity of the two phrases.

To address this aspect, we observe a word that has a direct semantic relationship with target word. The word is dependency head of target word. In the sentence, every word (except the dependency root word) has a dependency head, and it is related to the head. We attempt to verify the potential that context can influence popularity of sentence with dependency head. We created a Dependency Language

Model (DLM), which reflects the conditional probability of words and its dependency head.

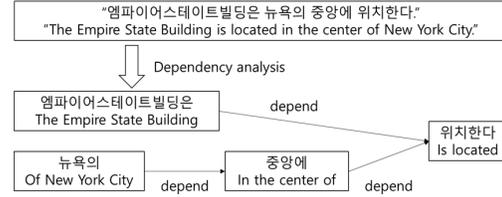


Figure 1: Example of dependency analysis in a Korean sentence.

We obtained the probability from a frequency investigation of a word pair after a dependency analysis. For example, the sentence “엠파이어스테이트빌딩은 뉴욕의 중앙에 위치한다. (The Empire State Building is located in the center of New York City.)” can be disassembled into {“엠파이어스테이트빌딩은 (The Empire State building)” → “위치한다 (is located)”}, {“뉴욕의 (of New York City)” → “중앙에 (in the center)”} and {“중앙에 (in the center)” → “위치한다 (is located)”}. This process is represented in Figure 1. Then, we investigated the conditional probabilities of those pairs. Thus, we calculate the conditional probability of words pairs as a unit of DLM. In addition, we applied morpheme analysis for the reasons described in Section 3.1. We used the dependency analyzer created by ETRI.

$$p(w|h_w) = \frac{CMC_{w,h_w}}{CMC_{h_w}} \quad (8)$$

Eq. (8) represents the conditional probability  $p(w|h_w)$  of word  $w$  and its head  $h_w$ .  $CMC_{w,h_w}$  is the number of co-occurrence of  $w$  and  $h_w$ . DLM is built by investigating all the dependency pairs of the corpus. Using the notion of DLM, we defined the context popularity  $g(w)$  as product of two words popularity (target word and its head) and their co-occurrence probability. It is represented in Eq. (9).

$$g(w) = f(w)p(w|h_w)f(h_w) \quad (9)$$

To measure sentence popularity with DLM, we calculate the context popularity of all dependency word pairs. This process is represented by the formula in Eq. (10). Lastly, to normalize the length  $n$

of sentences, we apply a logarithm and divide it by the number of dependency relationships  $n - 1$ .

$$D(s) = \prod_i^{n-1} f(w_i)p(w|h_{w_i})f(h_{w_i}) \quad (10)$$

$$G(s) = \frac{\sum_i^{n-1} \log g(w)}{n - 1} \quad (11)$$

Additionally, we can treat the word sense disambiguation (WSD) problem, too. For example, in the Korean language, the meaning of the noun “배 [bæ]” may be “pear” or “boat.” When we analyze the corpus to build CMC and DF, both nouns are treated as a single entity. This results in abnormally high statistical result scores, regardless of the actual frequency of each meaning. Using DLM, we can consider this problem with conditional probability. For example, the noun “배” means “pear” when its dependency head is *eat* or *squash*, and means “boat” if it is matched with *sail* or *steer*. We can infer the meaning and popularity of words in the context from its dependency head.

### 3.4 Measuring Total Popularity

We defined a popular sentence as one that contains words that are frequently used, appear in many documents, and contain frequent dependencies. In Eq. (12), we represent the sentence popularity  $H(s)$  by the sum of popularities from words  $F(s)$  and popularities from contexts  $G(s)$ . In the equation,  $\alpha$  and  $\beta$  are the weights of both popularities.

$$H(s) = \alpha F(s) + \beta G(s) \quad (12)$$

We can obtain Eq. (13) through substitution of Eq. (7) and (11) into Eq. (12).

$$H(s) = \alpha \frac{\sum_{i=1}^n f(w_i)}{n} + \beta \frac{\sum_{i=1}^{n-1} \log g(w_i)}{n - 1} \quad (13)$$

## 4 Experimental Setup

To evaluate how accurately our metric reflects the popularity that humans perceive when reading a sentence, we designed an experiment to measure correlation between human judgment and popularity.

### 4.1 Adoption of Dataset

To build the CMC, DF, and DLM, we need an appropriate corpus. When searching for a target corpus, the most important considerations were volume and ordinariness. Thus, we considered *Korean Wikipedia*<sup>2</sup> and *Modern Korean Usage Frequency Report (MKUFR)* (Hansaem, 2005) as suitable data sources. Korean Wikipedia is the Korean version of Wikipedia, the well-known collaborative online encyclopedia. Because it is written by public, we assume Korean Wikipedia contains moderately popular terms. Korean Wikipedia even offers a massive volume — more than 1.7 GB of data contained in over 700,000 documents. MKUFR is the result of research conducted by the National Institute of the Korean language from 2002 through 2005. They surveyed TF in publications printed between 1990 and 2002. Using Korean Wikipedia and MKUFR as a dataset, we built the CMC, DF, and DLM.

### 4.2 Human Evaluation Setup

We used a sentence set from TREC 2006 QA data<sup>3</sup> as test data. TREC (Text REtrieval Conference) is a conference focusing on information retrieval areas, and its dataset is widely used as a standard to evaluate the performance of information retrieval systems. We randomly selected 250 sentences from the TREC 2006 QA data and translated them into Korean by human translators. A paraphrase machine, based on Bannard and Callison-Burch’s algorithm (Bannard and Callison-Burch, 2005), was used to create machine-generated sentences from the translated TREC questions.

We employed five human judges (J1-5) to manually assess the popularity of 250 machine-generated sentences. The sentences were presented to the judges in random order. Each sentence was scored using a six-point scale. The instructions given to the judges were as follows.

*Popularity: Is the sentences linguistically popular?*

### 4.3 Inter-judge Correlation

Before evaluating our method, we used Pearson’s correlation coefficient to investigate the correlation between the human judges; these results are listed

<sup>2</sup><https://ko.wikipedia.org>

<sup>3</sup><http://trec.nist.gov/data/qa/>

	J1	J2	J3	J4	J5
J1	1	0.639	0.722	0.650	0.639
J2	0.639	1	0.582	0.496	0.645
J3	0.722	0.582	1	0.724	0.638
J4	0.650	0.496	0.724	1	0.536
J5	0.639	0.645	0.638	0.536	1

Table 1: Inter-judge correlation.

		J avg.	J1	J2	J3	J4	J5
Wiki	CMC	.45	.40	.37	.39	.33	.39
	CMCDF	<b>.58</b>	<b>.53</b>	<b>.51</b>	<b>.50</b>	<b>.40</b>	<b>.50</b>
UFR	CMC	.30	.28	.28	.24	.19	.27
	CMCDF	.43	.37	.40	.38	.27	.37

Table 2: Correlation between human judgment and popularity of each corpus.

in Table 1. Although J4 produced relatively poor results, correlations show a clear positive relationship between 0.49 and 0.72; excepting J4’s results, the correlation improved to between 0.58 and 0.72. These correlation scores can be regarded as fairly high, considering that we used a six-point scale and compared the results to similar results reported during the paraphrase evaluation (Liu et al., 2010). These high correlations confirm the effectiveness of our experimental design and explanation. We considered the reasons for J4’s relatively poor score when analyzing the results.

## 5 Experimental Result

### 5.1 Word Popularity

To measure sentence popularity with word popularity, we built language resources (CMC and DF) from each corpus: *Korean Wikipedia* and *Modern Korean usage frequency report*. Using Eq. (3) and Eq. (7), we calculated the popularity of each sentence. By comparing the performance of each corpus, we aim to identify the corpus that most accurately reflects public language usage. The correlations between our method and human judgments are listed in Table 2.

The results in Table 2 show, in general, clear positive linear correlations. The row labeled “Wiki” shows the results based on the Korean Wikipedia corpus; row “UFR” shows results based

on MKUFR. In particular, Wikipedia rather than MKUFR shows better performance, and CMC-DF (represented in Eq. (7)) shows better performance than CMC (Eq. (3)) only. We conclude that Korean Wikipedia reflects public language usage more accurately than MKUFR. Thus, we selected the Wikipedia corpus as the basis of our DLM, and conducted the experiment described below.

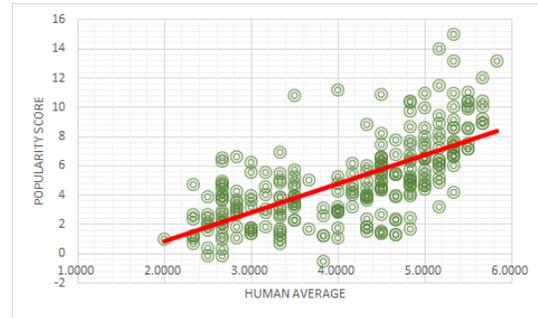


Figure 2: Scatter plot of popularity (un-optimized) versus human judgment (avg.).

### 5.2 Context Popularity

Through the previous experiment, we conclude that the *Wikipedia corpus* most accurately reflects public language usage. Thus, we built a DLM based on Wikipedia. Using Eq. (11), we measured context popularity based on dependency relationships. Lastly, we attempted to measure popularity by applying both word popularity and context popularity (this process is represented in Eq. (13)). In the Table 3, row “DLM” contains the results of applying context popularity (represented in Eq. (11)); row “Comb” contains the results of applying both word popularity and context popularity (represented in Eq. (13)). Figure 2 shows the average of human judgment scores plotted against the popularity derived from Eq. (13). Lastly, the results in row “Opt” show the result of optimization of the weight variables  $\alpha$  and  $\beta$  of Eq. (13). The optimization process will be discussed in Section 5.3. An interesting finding is that considering contexts alone is negatively correlated with human judgments. Nevertheless, when they are combined with word popularity, performance is improved. The Pearson correlation between popularity and human judgment is 0.77.

	Javg.	J1	J2	J3	J4	J5
CMC	.45	.40	.37	.39	.33	.39
CMCDF	.58	.53	.51	.50	.40	.50
DLM	-.20	-.23	-.17	-.15	-.17	-.22
Comb	.66	.62	.60	.56	.44	.62
<b>Opt</b>	<b>.77</b>	<b>.69</b>	<b>.60</b>	<b>.67</b>	<b>.45</b>	<b>.80</b>

Table 3: Correlation between human judgment and popularity of different models.

### 5.3 Weight Optimization

To derive optimal weight parameter  $\alpha$  and  $\beta$  in Eq. (13), we divide the experiment data into three sets: training, validation, and test. We divided the experiment data using a ratio of 3 : 1 : 1. Using a grid search, we identify the top ten parameter combinations. We set the scope of each parameter as integer  $[0, 100]$ . By applying those combinations to the validation set, we identify the optimal parameter pair; the parameter of CMC-DF( $\alpha$ ) is 35 and that of DLM( $\beta$ ) is 65. We verified the performance of our method with test set using the optimal parameter pair obtained from the validation set ( $\alpha : \beta = 35 : 65$ ).

### 5.4 Result Analysis

As in the Section 5.1, we investigated CMC-DF’s Pearson correlation with human judgments. Our basic concept started with term frequency; we built language resources (CMC) based on term frequency, and they showed a clear positive correlation of 0.45. In addition, we suggested that cases cannot be solved using only CMC. Thus, we applied DF, and we obtained an improved correlation of 0.58.

To measure popularity stemming from contextual information, we applied language modeling based on dependency relationships. Interestingly, DLM shows negative correlation by itself. However, when combined with CMC-DF, it improves correlation; the Pearson correlation between the combined model (CMC-DF-DLM) and human judgment is 0.66. We optimized the weight parameters through a grid search and avoid overfitting by dividing experiment data into three categories: training, validation, and test. The Pearson correlation between our popularity method and human judgment is 0.77. This

correlation is quite high, considering that the highest sentence-level Pearson correlation in the MetricMATR 2008 (Przybocki et al., 2009) competition was 0.68, which was achieved by METEOR; in contrast, BLEW showed a correlation of 0.45. When compared with the results of PEM (Liu et al., 2010), the sentence level correlation is also quite high.

Furthermore, we calculated the correlation between our method and each judge. Except for one judge, our method shows strong positive linear correlation with human judgments (between 0.60 and 0.80). Although the results produced by J4 were relatively poor, they still resulted in a clear positive correlation of 0.45.

### 5.5 Characteristics in Corpora and Judges

Table 2 shows that the CMC-DF based on Korean Wikipedia exhibit better performance than those based on MKUFR. The results from Section 5.1 became our grounds for concluding that Korean Wikipedia reflects public language usage more accurately than MKUFR. We believe the reasons are as follows.

- *Wikipedia* is written by the public.  
Modern Korean usage frequency report is based on publications written by experts such as writers, journalists, novelists, etc.
- *Wikipedia* is written in real time.  
Modern Korean usage frequency report was created in 2005 and analyzed publications printed between 1990 and 2002.

In Table 1, 2 and 3, we note that J4’s results show relatively low correlation with the results from other judges and the results from our methods. To reveal the reason, we analyzed their answer sheets. Table 4 shows statistical characteristic of human judgments. For each judge’s decision,  $\mu$  is the average score,  $\sigma$  represents the standard deviation, and *min* and *max* represent the lowest and highest values, respectively, in the range of responses. The salient point in Table 4 is that J4 assigned scores in a range of only  $[2, 5]$  while others used the entire scale  $[1, 6]$ .

### 5.6 Discussion and Future Work

As shown in Table 2 and 3, our method shows strong correlations with human judgments, even in

	Javg.	J1	J2	J3	J4	J5
$\mu$	4.11	3.35	4.69	3.74	4.41	3.02
$\sigma$	0.99	1.21	1.05	1.86	0.91	1.31
<i>min</i>	2.00	1	1	1	<b>2</b>	1
<i>max</i>	5.83	6	6	6	<b>5</b>	6

Table 4: Comparison of statistical characteristics of human judgments.

cases in which differences exist between individuals. Further, there is a clear improvement in correlation when the additional notions of document frequency and context are applied. In this experiment, our method showed the potential for measuring popularity by involving the contextual information; so far, we have considered only one word that has direct semantic relationship with target word, namely, the dependency head. The extension of contextual information will be addressed in future works.

Our method has a limitation due to lexical features. We cannot accommodate syntax-level popularity measures, such as the order of words. Because Korean is affiliated with agglutinative languages, there is no grammatical or semantic meaning related to the order of words; in sentences, empty morphemes decide the role of content morphemes. However, for readers and service consumers, the order of words can convey different impressions. This is an extension of the characteristics we aim to measure using popularity. These types of syntactic factors will be addressed in future works.

J4 showed relatively low Pearson correlation performance, breadth of improvement, and inter-judge correlation. To explain these, we developed two hypotheses. The first is that J4 assigned scores in a range of only [2, 5] while others used the entire scale [1, 6]. When conducting an experiment using the Likert scale, it is common for judges to avoid extreme estimations. This can reduce the sensitivity of the results. Low inter-judge correlation supports this hypothesis. The second is that he had a different standard of popularity. As mentioned previously, popularity is very subjective sense, and we focused on popularity stemming from lexical factors. If he followed different rules than other judges, the relatively low performance can be explained. Low im-

provement breadth per application of additional factors supports this hypothesis.

In aspects of application, we consider popularity as a method to reflect the style of sentences produced by MT or paraphrasing methods. Popularity is a type of combination of weighted probabilities. This means generating a possibility under a corpus that accurately reflects a target. In this paper, the target of the corpus was public language usage. However, if we secure various corpora that each reflects different targets, they can be used as classifiers to find the author of the source sentences.

Further, resources (CMC, DF, and DLM) can be used for generation module of MT or paraphrase system to reflect the specificity of the author. The target of corpus can be time, author, topic, or other factors. For example, assume that we have obtained diverse corpora from various authors, and one of the authors, “Murakami Haruki,” writes a new novel. A MT system containing the popularity module and language resources can identify the novel’s author and apply his style using the language resources from the Murakami’s corpus in generation stage.

## 6 Conclusion

In this paper, we proposed a novel notion, popularity, to consider the consumers’ satisfaction from functional quality. We defined a popular sentence as one that contains words that are frequently used, appear in many documents, and contain frequent dependencies. To measure the popularity, we began with term frequency, and then applied the concepts of document frequency and context to complement features that term frequency cannot cover. We conducted a human evaluation and measured the popularity for machine-generated sentences. In our experiment, we showed strong Pearson correlation coefficients between popularity and human judgment. To the best of our knowledge, our method is the first automatic sentence popularity evaluator based on term occurrences and contextual information.

## ACKNOWLEDGMENTS

This work was supported by ICT R&D program of MSIP/IITP. [R0101-15-0062, Development of Knowledge Evolutionary WiseQA Platform Technology for Human Knowledge Augmented Services]

## References

- Satanjeev Banerjee and Alon Lavie. 2005. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *Proceedings of the ACL workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, volume 29, pages 65–72.
- Colin Bannard and Chris Callison-Burch. 2005. Paraphrasing with bilingual parallel corpora. In *Proceedings of the 43rd Annual Meeting on ACL*, pages 597–604.
- Regina Barzilay and Lillian Lee. 2002. Bootstrapping lexical choice via multiple-sequence alignment. In *Proceedings of the ACL-02 conference on EMNLP-Volume 10*, pages 164–171.
- Chris Callison-Burch, Cameron Fordyce, Philipp Koehn, Christof Monz, and Josh Schroeder. 2007. (meta-) evaluation of machine translation. In *Proceedings of the Second Workshop on StatMT*, pages 136–158.
- Chris Callison-Burch, Trevor Cohn, and Mirella Lapata. 2008. Parametric: An automatic evaluation metric for paraphrasing. In *Proceedings of the 22nd Coling*, pages 97–104.
- Chris Callison-Burch. 2008. Syntactic constraints on paraphrases extracted from parallel corpora. In *Proceedings of the Conference on EMNLP*, pages 196–205.
- Luis Casaló, Carlos Flavián, and Miguel Guinalú. 2008. The role of perceived usability, reputation, satisfaction and consumer familiarity on the website loyalty formation process. *Computers in Human Behavior*, 24(2):325–345.
- David L Chen and William B Dolan. 2011. Collecting highly parallel data for paraphrase evaluation. In *Proceedings of the 49th Annual Meeting on the ACL*, pages 190–200.
- Kim Hansaem. 2005. Modern korean usage frequency report. <https://books.google.co.kr/books?id=umhKAQAIAAJ>.
- Vladimir I Levenshtein. 1966. Binary codes capable of correcting deletions, insertions, and reversals. In *Soviet Physics Doklady*, volume 10, pages 707–710.
- Chang Liu, Daniel Dahlmeier, and Hwee Tou Ng. 2010. Pem: A paraphrase evaluation metric exploiting parallel texts. In *Proceedings of EMNLP*, pages 923–932.
- Hans Peter Luhn. 1957. A statistical approach to mechanized encoding and searching of literary information. *IBM Journal of Research and Development*, 1(4):309–317.
- Philip M McCarthy, Rebekah H Guess, and Danielle S McNamara. 2009. The components of paraphrase evaluations. *Behavior Research Methods*, 41(3):682–690.
- Banwari Mittal and Walfried M Lassar. 1998. Why do customers switch? the dynamics of satisfaction versus loyalty. *Journal of Services Marketing*, 12(3):177–194.
- Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting on ACL*, pages 311–318.
- Mark Przybocki, Kay Peterson, Sébastien Bronsart, and Gregory Sanders. 2009. The nist 2008 metrics for machine translation challenge—overview, methodology, metrics, and results. *Machine Translation*, 23(2-3):71–103.
- Gerard Salton and Christopher Buckley. 1988. Term-weighting approaches in automatic text retrieval. *Information Processing & Management*, 24(5):513–523.
- Matthew G Snover, Nitin Madnani, Bonnie Dorr, and Richard Schwartz. 2009. Ter-plus: paraphrase, semantic, and alignment enhancements to translation edit rate. *Machine Translation*, 23(2-3):117–127.
- Robert A Wagner and Michael J Fischer. 1974. The string-to-string correction problem. *Journal of the ACM (JACM)*, 21(1):168–173.

# Acquiring distributed representations for verb-object pairs by using word2vec

Miki Iwai<sup>\*1</sup>, Takashi Ninomiya<sup>\*2</sup>, Kyo Kageura<sup>\*3</sup>

Graduate School of Interdisciplinary Information Studies, The University of Tokyo<sup>\*1</sup>

Graduate School of Science and Engineering, Ehime University<sup>\*2</sup>

Graduate School of Education, The University of Tokyo<sup>\*3</sup>

1156553643@mail.ecc.u-tokyo.ac.jp, ninomiya@cs.ehime-u.ac.jp, kyo@p.u-tokyo.ac.jp

## Abstract

We propose three methods for obtaining distributed representations for verb-object pairs in predicated argument structures by using word2vec. Word2vec is a method for acquiring distributed representations for a word by retrieving a weight matrix in neural networks. First, we analyze a large amount of text with an HPSG parser; then, we obtain distributed representations for the verb-object pairs by learning neural networks from the analyzed text. We evaluated our methods by measuring the MRR score for verb-object pairs and the Spearman's rank correlation coefficient for verb-object pairs in experiments.

## 1 Introduction

Natural language processing (NLP) based on corpora has become more common thanks to the improving performance of computers and development of various corpora. In corpus-based NLP, word representations and language statistics are automatically extracted from large amounts of text in order to learn models for specific NLP tasks. Complex representations of words or phrases can be expected to yield a precise model, but the data sparseness problem makes it difficult to learn good models with them; complex representations tend not to appear or appear only a few times in large corpora. For example, the models of statistical machine translation are learned from various statistical information in monolingual corpora or bilingual corpora. However, low-frequency word representations are not learned well, and consequently, they are processed as unknown words, which causes mistranslations. It is

necessary not only to process NLP tasks by matching surface forms but to generalize the language representations into semantic representations.

Many approaches represent words with vector space models so that texts can be analyzed using semantic representations for individual words or multi-word expressions. These methods can be classified into two approaches: the word occurrence approach and the word co-occurrence approach. The word occurrence approach includes Latent Semantic Analysis (LSA) (Landauer and Dumais, 1997), Probabilistic LSA (PLSA) (Hofman, 1999) and Latent Dirichlet Allocation (LDA) (Blei et al., 2003), which acquire word representations from the distributions of word frequencies in individual documents (a word-document matrix). Recently, many researchers have taken an interest in the word co-occurrence approach, including distributional representations and neural network language models (Mikolov et al., 2013a; Mikolov et al., 2013b; Mnih and Kavukcuoglu, 2013; Pennington et al., 2014). The word co-occurrence approach uses statistics of the context around a word. For example, the distributional representations for a word are defined as a vector that represents a distribution of words (word frequencies) in a fixed-size window around the word. The neural network language models, including word2vec (Mikolov et al., 2013a; Mikolov et al., 2013b), GloVe (Pennington et al., 2014) and vector Log-Bilinear Language model (vLBLE) (Mnih and Kavukcuoglu, 2013), generate distributed representations, which are dense and low-dimensional vectors representing word meanings, by learning a neural network that solves a pseudo-task of predict-

ing a word given its surrounding words. Word2vec is preferred in NLP because it learns distributed representations very efficiently. Neural network language models have semantic compositionality for word-word relations by calculating vector representations; e.g., ‘king’ - ‘man’ + ‘woman’ is close to ‘queen.’ However, they acquire the distributed representations for a word, not phrase structures such as verb and object pairs. It is necessary to obtain representations for phrases or sentences to be used as natural language representations.

We devised three methods for acquiring distributed representations for verb-object pairs by using word2vec. We experimentally verified that the distributed representations of different verb and object pairs have the same meaning. We focused on verb-object pairs consisting of verbs whose meaning is vague, such as light-verbs, e.g., the ‘do’ and ‘dishes’ pair in “do dishes”. The following two sentences are examples that have similar meanings but whose phrase structures are different.

1. I wash the dishes.
2. I do the dishes.

The representations for the verb-object pairs in the first sentence is “wash(dishes),” and those for the second sentence is “do(dishes)” with the light verb ‘do’. Despite the difference between the representations of these sentences, they have the same meaning “I wash the dishes.” As such, there are various sentences that have the same meaning, but different representations. We examined the performance of each method by measuring the distance between distributed representations for verb-object pairs (‘do’ and ‘dishes’ pair) and those for the corresponding basic verb (‘wash’) or predicated argument structures (“wash(dishes)”). We also experimentally compared the previous methods and ours on the same data set used in (Mitchell and Lapata, 2008).

## 2 Related work

There are many methods for acquiring word representations in vector space models. These methods can be classified into two approaches: the word occurrence approach and the word co-occurrence approach.

The word occurrence approach, including Latent Semantic Analysis (LSA) (Landauer and Dumais, 1997), Probabilistic LSA (PLSA) (Hofman, 1999) and Latent Dirichlet Allocation (LDA) (Blei et al., 2003), presupposes that distributions of word frequencies for each document (a word-document matrix) are given as input. In the word frequency approach, word representations are learned by applying singular value decomposition to the word-document matrix in LSA, or learning probabilities for hidden variables in PLSA or LDA. However, in the word frequency approach, the word frequencies for a document are given as a bag of words (BoW), and consequently, the information on the word order or phrase structure is not considered in these models.

The co-occurrence frequency approach, including distributional representations and neural network language models, uses statistics of the context around a word. The distributional representations for a word  $w$  are defined as a vector that represents the distribution of words (word frequencies) in a fixed-size window around word  $w$ , or the distribution of dependencies of word  $w$ , following the distributional hypothesis (Firth, 1957). Alternatively, neural network language models, including word2vec (Mikolov et al., 2013a; Mikolov et al., 2013b), GloVe (Pennington et al., 2014) and the vector Log-Bilinear Language model (vLBLE) (Mnih and Kavukcuoglu, 2013), generate dense and low-dimensional vectors that represent word meanings by learning a neural network that solves a pseudo-task in which the neural network predicts a word given surrounding words. After the training of the neural network on a large corpus, the word vector for  $w$  is acquired by retrieving the weights between  $w$  and the hidden variables in the neural network (Bengio et al., 2003; Collobert and Weston, 2008). Word2vec is preferred in NLP because it learns distributed representations very efficiently. The conventional methods for neural network language models take several weeks to learn their models on tens of millions sentences in Wikipedia (Collobert et al., 2011). It is likely possible for word2vec to reduce the calculation time dramatically. However, these models basically learn word-to-word relations, not phrase or sentence structures.

When we make distributed representations for phrases or sentences, it is necessary to generate con-

stitutive distributed representations for phrases or sentences based on the principle of compositionality. Mitchell and Lapata (2008) and Mitchell and Lapata (2010) proposed the add model, which generates distributed representations for phrase structures, whereas Goller and Küchler (1996), Socher et al. (2012) and Tsubaki et al. (2013) proposed Recursive Neural Network (RNN) models for phrase structures. Recently, new models based on tensor factorization have been proposed (Baroni and Zamparelli, 2010; Grefenstette and Sadrzadeh, 2011; Kartsaklis et al., 2012).

The add model is a method to generate distributed representations for phrase structures or multi-word expressions by adding distributed representations for each word that constitutes the phrase structure. However, the word order and syntactic relations are lost as a result of the adding in the model. For example, suppose that we have the distributed representations for a verb, a subject and an object. The result of adding the distributed representations is the same if we change the order of the subject and the object. For example, consider the distributed representations for the following two sentences.

- The girl gave a present.
- A present gave the girl.

The distributed representations for these sentences are as follows.

$$\begin{aligned}
 & v(\text{the}) + v(\text{girl}) + v(\text{gave}) + v(\text{a}) + v(\text{present}) \\
 = & v(\text{a}) + v(\text{present}) + v(\text{gave}) + v(\text{the}) + v(\text{girl})
 \end{aligned}$$

where  $v(w)$  is the distributed representations for word  $w$ . It is necessary for the models to be sensitive to the word order to make a difference between these sentences. To solve these problems, various approaches have been proposed. For example, a method that adds weights to verbal vectors appearing ahead or one that assigns word-order numbers to  $n$ -grams was proposed. RNNM can acquire distributed representations for one sentence using RNN and a given syntactic tree (Socher et al., 2011; Socher et al., 2012). RNNM makes use of syntactic trees of sentences, as shown in Figure 1. It calculates a distributed representation for the parent node from

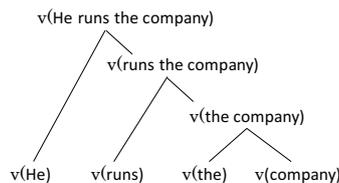


Figure 1: RNNM structure with syntax tree

the distributed representations for the child nodes in the syntactic trees. However, it uses only the skeletal structures of the syntactic trees; category and subject information in the syntactic trees are not used. Hashimoto et al. (2014) proposed a new method that acquires distributed representations for one sentence with information on words and phrase structures by using the parse trees generated by an HPSG parser called Enju.

Tensor factorization is a method that represents word meaning with not only vectors but also matrices. For example, a concept ‘car’ has many attributes such as information about color, shape, and functions. It seems to be difficult to represent phrases or sentences with a fixed-size vector because many concepts can appear in a sentence and each concept has its own attributes. Baroni and Zamparelli (2010) tried to represent attribute information of each word as a product of a matrix and a vector. Grefenstette and Sadrzadeh (2011) followed this approach and proposed new method that obtains the representations of verb meaning as tensors. Kartsaklis et al. (2012) proposed a method that calculates representations for sentences or phrases containing a subject, a verb and an object, based on Grefenstette and Sadrzadeh (2011)’s method. Recently, three dimensional tensors have been used for representing the relations of a subject, a verb and an object (de Cruys, 2009; de Cruys et al., 2013).

### 3 Word2vec

Word2vec (Mikolov et al., 2013a; Mikolov et al., 2013b) is the method to obtain distributed representations for a word by using neural networks with one hidden layer. It learns neural network models

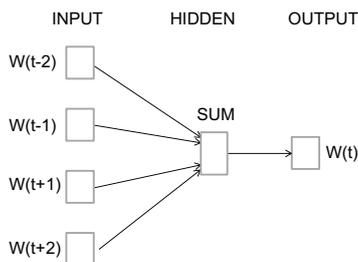


Figure 2: CBOw

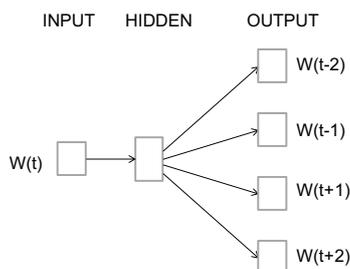


Figure 3: Skip-gram

from large texts by solving a pseudo-task to predict a word from surrounding words in the text. The word weights between the input layer and hidden layer are extracted from the network and become the distributed representation for the words. Mikolov et al. proposed two types of network for word2vec, the Continuous Bag-of-words (CBOw) model and the Skip-gram model.

### 3.1 CBOw model

Figure 2 shows the CBOw model’s network structure. The CBOw model is a neural network with one hidden layer, where the input is surrounding words  $w_{t-k}, \dots, w_{t-1}, w_{t+1}, \dots, w_k$ , and the output is  $w_t$ . The input layer and output layer are composed of nodes, each of which corresponds to a word in a dictionary; i.e., input and output vectors for a word are expressed in a 1-of- $k$  representation. The node values in the hidden layer are calculated as the sum of the weight vectors of the surrounding words  $w_{t-k}, \dots, w_{t-1}, w_{t+1}, \dots, w_k$ .

### 3.2 Skip-gram model

Figure 3 shows the Skip-gram model’s network structure. The Skip-gram model is a neural network with one hidden layer in which a 1-of- $k$  vector for word  $w_t$  is given as an input

and 1-of- $k$  vectors for the surrounding words  $w_{t-k}, \dots, w_{t-1}, w_{t+1}, \dots, w_k$  are output.

## 4 Proposed methods

This section explains the proposed methods to obtain the distributed representations for verb-object pairs by using word2vec. First, we explain the baseline for comparison of the proposed methods. Then, we describe the proposed methods.

### 4.1 Baseline method

The baseline method is the add model using word2vec. Word2vec is first trained with a large amount of text; then, distributed representations for each word are obtained. For example, the vector for “read” is obtained as “read = (1.016257, -1.567719, -1.891073, ..., 0.578905, 1.430178, 1.616185)”. Distributed representations for a verb-object pair are obtained by adding the vector for the verb and the vector for the object.

### 4.2 Method 1

The CBOw model of word2vec is learned in a pseudo-task that predicts a word from surrounding words in the text. Thus, we expect that distributed representations for verb-object pairs can be acquired when the object is put near the verb. A large amount of training text is parsed by Enju, and new training text data is generated by inserting the object just after the verb for all verb-object pairs appearing in the corpus as follows.

(original) I did many large white and blue round dishes.

(modified) I **do dish** many large white and blue round dish.

Enju (Miyao et al., 2005; Miyao and Tsujii, 2005; Ninomiya et al., 2006) is a parser that performs high-speed and high-precision parsing and generates syntactic structures based on HPSG theory (Pollard and Sag, 1994), a sophisticated grammar theory in linguistics. In addition, Enju can generate predicate argument structures. The Stanford Parser (de Marneffe et al., 2006; Chen and D.Manning, 2014) is often used, but it can analyze only syntactic structures. Therefore, we used Enju, which can parse syntactic structures and predicate argument structures. In

Method 1, word2vec is trained from the new text data generated by using Enju’s results to augment objects near verbs in the text. Then, distributed representations for verb-object pairs are generated by adding the distributed representations for the verb and the distributed representations for the object.

### 4.3 Method 2

We expect that distributed representations for verb-object pairs can be obtained by training word2vec with text in which each verb is concatenated with its object for all verb-object pairs. For each verb  $v$  and object  $o$  pair,  $v$  is replaced with  $v : o$ , where  $v$  and  $o$  are concatenated into a single word using Enju’s result. The following shows an example of Method 2.

(original) I did many large white and blue round dishes.

(modified) I **do:dish** many large white and blue round dish.

Word2vec is learned using the new generated text, and distributed representations for verb-object pairs are acquired.

### 4.4 Method 3

The Skip-gram model is learned by solving a pseudo-task in which a word in the text is given as input, and the neural network predicts each surrounding word. It is likely that distributed representations for verb-object pairs can be acquired by providing the verb and its object to the neural networks at the same time when the input word is a verb. We performed the learning in Method 3 by using a new Skip-gram model wherein the verb-object pair is input to the neural networks when one of the input words is a verb.

Figure 4 shows the neural network model for Method 3. The model is trained from a large amount of text, and distributed representations for words are generated. Then, the distributed representations for verb-object pairs are acquired by summing the distributed representations for the verb and the distributed representations for the object in the same way as Method 1.

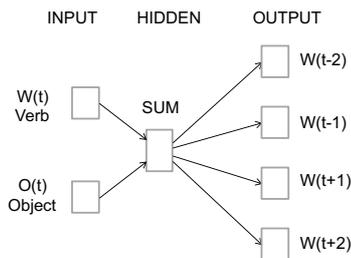


Figure 4: New Skip-gram model

## 5 Experiments and evaluations

We performed two experiments to evaluate the performance of Methods 1, 2, 3, and the baseline method. We used word2vec in the experiments for Methods 1, 2, and the baseline with the CBOW model option (-cbow 1) and a modified word2vec for Method 3 based on the Skip-gram model. In all methods, the maximum window size was 8 words (-window 8), the sample number for negative sampling was 25 (-negative 25), and we did not use hierarchical softmax (-hs 0). The number of nodes in the hidden layer was 200; i.e., the number of dimensions for the distributed representations was 200.

### 5.1 Experiment on light verb-object pairs

We performed an experiment on pairs of a light verb and an object. The training corpus consisted of the English Gigaword 4th edition (LDC2009T13, nyt\_eng, 199412 - 199908), Corpus of Contemporary American English (COCA), and Corpus of Historical American English (COHA). The size of the training corpus was about 200 million words.

We developed a data set that consists of 17 triples of a light verb, an object, and a basic verb. The basic verb is one that almost has the same meaning as the corresponding light-verb and object pair. Table 1 shows examples of the data set. The pairs were selected from “Eigo Kihon Doushi Katsuyou Jiten (The dictionary of basic conjugate verbs in English)” (Watanabe, 1998) and a web site<sup>1</sup>. The basic verbs were selected from “Eigo Kihon Doushi Jiten (The dictionary of basic verbs in English)” (Konishi, 1980).

We evaluated each method by measuring the

<sup>1</sup>web page (<http://english-leaders.com/hot-three-verbs/1/20/2015> reference)

Table 1: Examples of distributed representations for light verb-object pairs

verb-object pairs	basic verbs	examples
do-dish	wash	I do the dishes.
do-cleaning	clean	I'll do the cleaning.
do-nail	put, paint, dress	We do our hair, and then we do our nails.
do-laundry	wash	I'm doing the laundry.
have-lunch	eat	Let's have lunch.
have-tea	drink	Let's have some tea.
have-word	tell, talk, speak	I'd like to have a word with you.
make-call	call	I always get nervous whenever I make a call.
make-bed	clean, put, set	I make the bed.
hold-door	open	Hold the door.
hold-tongue	shut	Hold your tongue!
give-hand	help	Give me a hand with this box.
give-party	hold, have, throw	She is giving a party this evening.
give-news	report, present, announce	I will probably be able to give you good news.
finish-coffee	drink	He finished his coffee.
read-shakespeare	read	I read Shakespeare.
enjoy-movie	watch,see	Did you enjoy the movie?

mean reciprocal rank (MRR) score for each verb-object pair in the data set, supposing that the corresponding basic verb is the true answer for the pair. Given a verb-object pair, we calculated its MRR score as follows. First, we calculated the cosine distance between the verb-object pair and all basic verb candidates in the dictionary. Then, we ranked the basic verbs in accordance with the cosine measure. The candidates of the basic verbs were 385 words in the basic verb dictionary (Konishi, 1980).

### 5.2 Comparison with conventional methods

We also conducted experiments with the data set<sup>2</sup> provided by Mitchell and Lapata (2008). This set consists of triples (*pair1*, *pair2*, *similarity*), from

<sup>2</sup><http://homepages.inf.ed.ac.uk/s0453356/share>

Table 2: Results for light verb-object pairs (Average of MRR)

baseline	Method 1	Method 2	Method 3
0.27	0.35	0.37	0.31

which we used 1890 verb and object pairs. The semantic similarity scores in the data set are given manually and range between 1 (low similarity) to 7 (high similarity). There are three types of combinations for *pair1* and *pair2* in the data: adjective + noun, noun + noun, and verb + object. For example, the similarity score for “vast amount” and “large quantity” is 7, and the similarity score for “hear word” and “remember name” is 1. We calculated Spearman’s rank correlation coefficient on the “verb + object” part of this data set. The similarity scores for verb-object pair *pair1* and *pair2* were calculated using the cosine similarity between the vector for *pair1* and the vector for *pair2*. If a system achieved a higher correlation coefficient, this means that its judgment was similar to that of humans.

## 6 Results

### 6.1 Results for light verb-object pairs

Table 2 shows the average MRR score for each method. Method 2 achieved the best result. We consider that training with the text in which verb-object pairs were replaced with a single expression had a good effect on word2vec. Method 1 and Method 3’s similarities were also higher than those of the baseline method. Therefore, it can be considered that distributed representations for verb-object pairs that were sensitive to verb-object relations were acquired by improving the training data. However, Method 1 achieved a higher MRR than that of Method 3. We consider that this is because Method 3 learned the model from heterogeneous structures; i.e., the hidden layer in the neural networks received different signals depending on whether the input was a verb or not.

Table 3 shows the details of the experimental results. From the table, we can see that Method 1 outperforms Method 2 in many cases, although the average MRR of Method 2 is greater than that of Method 1. We think that this is because Method

Table 3: Details of the experiment

VO	baseline	Method 1	Method 2	Method 3
do-dish	0.03	0.08	1	0.07
do-cleaning	0.05	0.14	0.25	0.33
do-nail	0.02	0.02	0.38	0.06
do-laundry	0.17	0.07	0.09	0.14
have-lunch	1	1	0.2	0.33
have-tea	0.5	1	1	0.5
have-word	0.12	0.07	0.05	0.12
make-call	1	1	1	1
make-bed	0.02	0.04	0.03	0.05
hold-door	0.02	0.5	0.2	0.5
hold-tongue	0.01	0.01	0.005	0.01
give-hand	0.03	0.13	0.05	0.05
give-party	0.05	0.07	0.01	0.19
give-news	0.02	0.12	0.01	0.06
finish-coffee	0.5	0.5	1	0.5
read-shakespeare	1	1	1	1
enjoy-movie	0.11	0.13	0.02	0.38

2 achieved similarity 1 in some cases, and this increased the average MRR.

## 6.2 Comparison with conventional method

Table 4 shows the results of Methods 1, 2, and 3 and the baseline method using Skip-gram and CBOW with Mitchell and Lapata’s data set. Method 1 using CBOW and size 50 achieved the best result. The reason is the process of learning. The CBOW model predicts a word by adding the vectors of surrounding words. Therefore, Method 1 with the CBOW model predicts a word from the sum of the vectors for a verb and its object. Consequently, representations for verb-object pairs are consistent in the learning and generating processes.

Table 5 shows the comparison with other methods. BL, HB, KS, and K denote the results of the methods of Blacoe and Lapata (2012), Hermann and Blunsom (2013), Kartsaklis and Sadrzadeh (2013), and Kartsaklis et al. (2013). Kartsaklis and Sadrzadeh (2013) used the ukWaC corpus (Baroni et al., 2009), and the other methods used the British National Corpus (BNC). Word2vec is the result of Hashimoto et al. (2014). They used the POS-tagged BNC and trained 50-dimensional word vectors with the Skip-gram model. We believe that our methods can be improved by using POS-tagged texts.

Table 4: Results for verb-object pairs in Mitchell and Lapata’s data set (Spearman’s rank correlation coefficient)

Method	Option	Score
Base-line	CBOW, -size 50	0.323
Method1	CBOW, -size 50	0.329
Method2	CBOW, -size 50	0.233
Base-line	Skip-gram, -size 50	0.308
Method1	Skip-gram, -size 50	0.305
Method2	Skip-gram, -size 50	0.173
Method 3	Skip-gram, -size50	0.272
Base-line	CBOW, -size 200	0.321
Method1	CBOW, -size 200	0.328
Method2	CBOW, -size 200	0.201
Base-line	Skip-gram, -size 200	0.308
Method1	Skip-gram, -size 200	0.292
Method2	Skip-gram, -size 200	0.171
Method 3	Skip-gram, -size200	0.275

Table 5: Comparison with other methods

Method	Score
Method 1 with CBOW	0.329
BL w/ BNC	0.35
HB w/ BNC	0.34
KS w/ ukWaC	0.45
K w/BNC	0.41
Word2vec	0.42

## 7 Conclusion and future work

This paper proposed methods for obtaining distributed representations for verb-object pairs by using word2vec. We experimentally evaluated them in comparison with the baseline add method in terms of mean reciprocal rank and Spearman’s rank correlation. Method 2, which concatenates verbs with their objects in the text, achieved the best MRR score in the experiment on light verb-object pairs. Method 1, which puts objects nearby verbs, achieved the best correlation coefficient in the experiment on Mitchell and Lapata’s data set. We consider that the training text data in these experiments was too small. It is necessary to use a large amount of data to verify which method is best for obtaining distributed representations of verb-object pairs. Using a large amount of data and making comparisons

with RNNM and tensor factorization are left as future work.

## Acknowledgments

This work was supported by JSPS KAKENHI Grant-in-Aid for Scientific Research (B) Grant Number 25280084.

## References

- Marco Baroni and Roberto Zamparelli. 2010. Nouns are vectors, adjectives are matrices: Representing adjective-noun constructions in semantic space. In *proceedings of the Conference on the Empirical Methods in Natural Language Processing (EMNLP 2010)*, pages 1183–1193.
- Marco Baroni, Silvia Bernardini, Adriano Ferraresi, and Eros Zanchetta. 2009. The wacky wide web: A collection of very large linguistically processed web-crawled corpora. *proceedings of Language Resources and Evaluation Conference (LREC 2009)*, pages 209–226.
- Yoshua Bengio, Rejean Ducharme, Pascal Vincent, and Christian Jauvin. 2003. A neural probabilistic language model. *Journal of Machine Learning Research*, 3:1137–1155.
- William Blacoe and Mirella Lapata. 2012. A comparison of vector-based representations for semantic composition. In *proceedings of the Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 546–556.
- David M. Blei, Andrew Y. Ng, and Michael I. Jordan. 2003. Latent dirichlet allocation. *Journal of Machine Learning Research*, pages 993–1022.
- Danqi Chen and Christopher D. Manning. 2014. A fast and accurate dependency parser using neural networks. In *proceedings of the Conference on the Empirical Methods in Natural Language Processing (EMNLP 2014)*, pages 740–750.
- Ronan Collobert and Jason Weston. 2008. A unified architecture for natural language processing: Deep neural networks with multitask learning. In *proceedings of the International Conference on Machine Learning (ICML 2008)*, pages 160–167.
- Ronan Collobert, Jason Weston, Leon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. 2011. Natural language processing (almost) from scratch. *Journal of Machine Learning Research*, pages 2493–2537.
- Tim Van de Cruys, Thierry Poibeau, and Anna Korhonen. 2013. A tensor-based factorization model of semantic compositionality. In *proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics (NAACL 2013) : Human Language Technologies*, pages 1142–1151.
- Tim Van de Cruys. 2009. A non-negative tensor factorization model for selectional preference induction. In *proceedings of the Workshop on Geometrical Models of Natural Language Semantics*, pages 83–90.
- Marie-Catherine de Marneffe, Bill MacCartney, and Christopher D. Manning. 2006. Generating typed dependency parses from phrase structure parses. In *proceedings of Language Resources and Evaluation Conference (LREC 2006)*, pages 449–454.
- John R. Firth. 1957. A synopsis of linguistic theory 1930–55. *Studies in Linguistic Analysis*, pages 1–32.
- Christoph Goller and Andreas Küchler. 1996. Learning task-dependent distributed representations by back-propagation through structure. *International Conference on Neural Networks*.
- Edward Grefenstette and Mehrnoosh Sadrzadeh. 2011. Experimental support for a categorical compositional distributional model of meaning. In *proceedings of the Conference on the Empirical Methods in Natural Language Processing (EMNLP 2011)*, pages 1394–1404.
- Kazuma Hashimoto, Pontus Stenetorp, Makoto Miwa, and Yoshimasa Tsuruoka. 2014. Jointly learning word representations and composition functions using predicate-argument structures. In *proceedings of the Conference on the Empirical Methods in Natural Language Processing (EMNLP 2014)*, pages 1544–1555.
- Karl Moritz Hermann and Philip Blunsom. 2013. The role of syntax in vector space models of compositional semantics. In *Annual Meeting of the Association for Computational Linguistics*, pages 894–904.
- Thomas Hofman. 1999. Probabilistic latent semantic analysis. In *Uncertainty in Artificial Intelligence*.
- Dimitri Kartsaklis and Mehrnoosh Sadrzadeh. 2013. Prior disambiguation of word tensors for constructing sentence vectors. In *proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP 2013)*, pages 1590–1601.
- Dimitri Kartsaklis, Mehrnoosh Sadrzadeh, and Stephen Pulman. 2012. A unified sentence space for categorical distributional-compositional semantics: Theory and experiments. In *proceedings of the International Conference on Computational Linguistics (Coling 2012)*, pages 549–558.
- Dimitri Kartsaklis, Mehrnoosh Sadrzadeh, and Stephen Pulman. 2013. Separating disambiguation from composition in distributional semantics. In *proceedings of the Conference on Natural Language Learning (CoNLL 2013)*, pages 114–123.

- Tomohichi Konishi. 1980. *Eigo Kihon Doushi Jiten (The dictionary of basic verbs in English)*. Kenkyusha publication.
- Thomas K. Landauer and Susan T. Dumais. 1997. A solution to plato's problem: The latent semantic analysis theory of the acquisition, induction, and representation of knowledge. *Psychological Review*, 104(2):211–240.
- Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013a. Efficient estimation of word representations in vector space. In *proceedings of workshop at the International Conference on Learning Representations (ICLR 2013)*.
- Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeffrey Dean. 2013b. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems 26*, pages 3111–3119.
- Jeff Mitchell and Mirella Lapata. 2008. Vector-based models of semantic composition. In *proceedings of the Association for Computational Linguistics (ACL 2008)*, pages 236–244.
- Jeff Mitchell and Mirella Lapata. 2010. Composition in distributional models of semantics. *Cognitive sentence*, 34(8):1388–1439.
- Yusuke Miyao and Jun'ichi Tsujii. 2005. Probabilistic disambiguation models for wide-coverage hpsg parsing. In *proceedings of the Association for Computational Linguistics (ACL 2005)*, pages 83–90.
- Yusuke Miyao, Takashi Ninomiya, and Jun'ichi Tsujii, 2005. *Keh-Yih Su, Jun'ichi Tsujii, Jong-Hyeok Lee and Oi Yee Kwong (Eds.), Natural Language Processing - IJCNLP 2004 LNAI 3248*, chapter Corpus-oriented Grammar Development for Acquiring a Head-driven Phrase Structure Grammar from the Penn Treebank, pages 684–693. Springer-Verlag.
- Andriy Mnih and Koray Kavukcuoglu. 2013. Learning word embeddings efficiently with noise-contrastive estimation. In *Conference on Neural Information Processing System 2013*, pages 2265–2273.
- Takashi Ninomiya, Takuya Matsuzaki, Yoshimasa Tsuruoka, Yusuke Miyao, and Jun'ichi Tsujii. 2006. Extremely lexicalized models for accurate and fast hpsg parsing. In *proceedings of the Conference on the Empirical Methods in Natural Language processing (EMNLP 2006)*.
- Jeffery Pennington, Richard Socher, and Christopher D. Manning. 2014. Glove: Global vectors for word representation. In *proceedings of the Conference on the Empirical Methods in Natural Language processing (EMNLP 2014)*, pages 1532–1543.
- Carl Pollard and Ivan A. Sag. 1994. Head-driven phrase structure grammar. *University of Chicago Press*.
- Richard Socher, Eric H. Huang, Jeffrey Pennington, Christopher D. Manning, and Andrew Y. Ng. 2011. Dynamic pooling and unfolding recursive autoencoders for paraphrase detection. In *Advances in Neural Information Processing Systems 24*, pages 801–809.
- Richard Socher, Brody Huval, Christopher D. Manning, and Andrew Y. Ng. 2012. Semantic compositionality through recursive matrix-vector spaces. In *proceedings of the Conference on the Empirical Methods in Natural Language processing (EMNLP 2012)*, pages 1201–1211.
- Masashi Tsubaki, Kevin Duh, Masashi Shimbo, and Yuji Matsumoto. 2013. Modeling and learning semantic co-compositionality through prototype projections and neural networks. In *proceedings of the Conference on the Empirical Methods in Natural Language processing (EMNLP 2013)*, pages 130–140.
- Miyoko Watanabe. 1998. *Eigo Kihon Doushi Katsuyou Jiten (The dictionary of basic conjugate verbs in English)*. Nagumo phoenix publication.

# Dependency Parsing for Chinese Long Sentence: A Second-stage Main Structure Parsing Method

**Bo Li**

School of informatics  
University of Edinburgh  
11 Crichton St, Edinburgh  
EH8 9LE, UK  
libo.whu@gmail.com

**YunFei Long**

School of Computer Science  
Nanjing Normal University  
No 1, Wen Yuan Road, Nan-  
jing, China  
893997052@qq.com

**WeiGuang Qu**

School of Computer Science  
Nanjing Normal University  
No 1, Wen Yuan Road,  
Nanjing, China  
wgqu\_nj@163.com

## Abstract

This paper explores the problem of parsing Chinese long sentences. Inspired by human sentence processing, a second-stage parsing method, referred as main structure parsing in this paper, are proposed to improve the parsing performance as well as maintaining its high accuracy and efficiency on Chinese long sentences. Three different methods have attempted in this paper and the result shows that the best performance comes from the method using Chinese comma as the boundary of the sub - sentence. According to our experiment about testing on the Chinese dependency Treebank 1.0 data, it improves long dependency accuracy by around 6.0% than the baseline parser and 3.2% than the previous best model.

## 1 Introduction

In recent years, the transition-based dependency parsing has been a hot research topic in Chinese parsing because of suitable to Chinese grammar profile and its linear scale time complexity. (Zhou, 2000) (Nivre and McDonald, 2008). However, although transition-based dependency parsing research has made great progress with the state-of-art performing at around 86% accuracy (Nivre et al., 2011), it still faces some problems when parsing Chinese long sentences.

First, the parser performance decreases when the length of input Chinese sentence increases. In other words, it cannot parse Chinese long sentences as accurate as short ones. As a result, if there are more long sentences in the input sentences, the overall accuracy will be affected significantly. The experiments in this paper on sentences of different length ranges show that the overall accuracy will decrease more than 1% when the length of input sentences is more than 50. This phenomenon is not only present in Chi-

nese long sentences, but also found during parsing research of other languages such as English and French (Candito et al 2012).

The second problem is that long sentences always contain global ambiguities, and the inaccuracies on long sentences can lead to a very different understanding of a sentence. While the short sentences have more local ambiguity and inaccuracies on short sentences normally, only cause misunderstanding on details. This is because long sentences tend to contain more details about semantic and discourse information compared with short sentences. Those details confuse parsers and prevent them from finding out what the correct structure of the long sentence.

Although the reasons that should be responsible for the performance decrease in parsing long sentences are still controversial, a common explanation is that there are some rarely seen features in long sentences causes the degraded performance (Candito et al., 2012).

Unfortunately, these features cannot be learnt by transition-based parser via increasing the scale of training corpus, because the idea of the transition-based dependency parsing methods is to process a sentence incrementally, some global information from those input sentences has been neglected during the process. Attempts to include that global information in transition-based dependency parsing have been made in past years (Nivre and McDonald, 2008; Nivre et al., 2011), but those methods always have to make a tradeoff between accuracy and efficiency. What this paper tries to propose is a parsing method that achieves better performance when parsing Chinese long sentences and freezes the  $O(n)$  time complexity simultaneously.

The fact that humans can understand a long sentence correctly even when some words are unknown is quite inspiring. It implies that not all words are equally important in terms of understanding a sentence. Some words carry more syn-

tactic and semantic information than others during people sentence understanding. Errors in recognizing those words may lead to understanding problems.

This is also true for dependency parsers. The reason why it cannot parse long sentence accurately is it does not distinguish those words from all words in a long sentence. In short sentence case, those words are always can be found because the pattern between those words is limited, which means a large training corpus can almost cover all the patterns between words, but that does not work well in long sentences. On one hand, as the input sentence gets longer, the possible combinations between words will outnumber the patterns can be found in the training corpus. On the other hand, there will be sentence-level instead of only word-level combination in a long sentence, which is beyond the transition-based parsing mode.

Therefore, this paper proposes a two-stage parsing method to help parsers find out those important words in sentences and use the information to improve parsing performance with out at the expense of time complexity.

## 2 Related Work

Dominating dependency-parsing models can be categorized into three families: graph-based models (Eisner, 1996; McDonald et al., 2005; Mc-Donald and Pereira, 2006; Wang et al., 2007; Zhang and Clark, 2008), transition-based models (Yamada and Matsumoto, 2003; Nivre and Scholz, 2004) and hybrid models (Sagae and Lavie, 2006; Nivre and McDonald, 2008; Zhang and Clark, 2008).

The advantage of the graph-based parsing is that it processes the input sentence as a whole. In other words, it takes global information of the input sentence into consideration, which gives it a higher accuracy on average than other models (Nivre, 2007) However, because of adopting global information, the efficiency of graph-based parsing models are comparatively lower ( $O(n^2)$ ) as the searching space is much larger.

By contrast, transition-based model, which is also referred as action-based parsing model, significantly outperform in efficiency. The transition-based parsing is essentially a discriminative algorithm which processes words incrementally. According to (Nivre and McDonald, 2008), transition-based parsing gives time complexity as low as  $O(n)$  (projective situation).

In Chinese dependency parsing research, transition-based parsing is a preferable choice because it suits better with the syntax of Chinese (Lai and Huang, 1994; Lai et al., 2001; Wang, William Yang, et al., 2014). Compared with English dependency parsing, Chinese dependency parsing is slightly underperformed. That is partially because there are a few widely used Chinese dependency corpus. The Penn Chinese TreeBank (CTB) is a promising choice. However, it is still not complete enough compared with that in English. For performance evaluation, Nivre (2011) provide a widely accepted comparison result, according to this paper, the state-of-art performance of Chinese dependency parsing is around 86.0% in unlabeled attachment scores (UAS).

Some recent research on improving the Chinese parsing performance by introducing multiple layer parsing approach (Ping Jian, et al., 2009) has been made, but it does not consider Chinese features. Zhenghua et al (2010) proposed the idea of using punctuation to help improving parsing, which also been discussed in this paper. However, the major difference is that punctuation is just one perspective of the framework proposed in this paper. In addition, this paper achieves a better performance compared with previous works.

## 3 A New Framework

### 3.1 Framework Design and Parsing Process

As previous discussion, the key to parse long sentences accurately is to find out the words that carry structural information about the sentence, which named as the main structure words in this paper. However, the challenge is that normal transition-based parsing methods cannot find out those main structure words because of lack of global information. In this circumstance, we select the output of a transition-based parsing method, which contains candidate features for main structure word recognition, after main structure word recognition, a second transition-based parser, which trained in a special corpus, introduced to adjust the dependencies between those main structure words. This second-stage parsing method referred as main structure parsing.

The purpose of the first parsing stage is to find out main structure words. In the first parsing stage, the baseline parser parses the input sentence in the normal way. From the output of baseline parser, the information for finding out main structure words extracted by following cer-

tain steps in the framework, and then the information is pass into the next parsing stage.

The information obtained from baseline parser is:

**Short Dependencies:** The short dependencies are normally from words that occur in the same sub-sentence (sub-sentence is the part between two punctuations in a sentence; a long sentence normally consists of multiple sub-sentences). The structure of these dependencies tends to be less complicated, and traditional transition-based parser can achieve over 90% accuracy on these dependencies. As the accuracy for these short dependencies is high, they are assumed as correct dependencies within the sub-sentence. Therefore, in the next stage, the main structure parser can only focus on these long dependencies, which is the key idea of the framework.

**Long Dependencies:** The long dependencies normally occur between words from different sub-sentences. The words that carry long dependencies are potential main structure words; normally they determine the global structure of the whole sentence. However, as the dependencies between main structure words are much longer than normal dependencies, traditional transition-based parsers are inaccurate on them. Given this, this paper uses a specially trained parser to re-parse the long dependencies regardless of the short dependencies. The result can be merged with short dependencies from first stage parsing through a voting scheme.

**Other Information:** Including the length of the input sentence, the number of sub-sentences, etc.

The challenge after obtaining the three kinds of information is how to distinguish actual main structure words from these potential ones. Three different methods are proposed and discussed in the paper in Chapter 3.2.

The goal of the second parsing stage is to find out correct dependencies between the main structure words. From previous discussion, the reason why a transition-based parser cannot parse main structure words correctly in long sentence is that syntactically redundant detail brings significant ambiguity. Therefore, in this stage, those details are ignored temporarily and only main structure words are processed. Obviously, a parser trained in normal corpus is not able to parse main structure words directly. The parser used in the second stage will be trained in a special corpus that only contains main structure words. As there is no available corpus like this, this paper adopts a special training corpus produced by the automat-

ic main structure words extraction method introduced in Chapter 3.2.

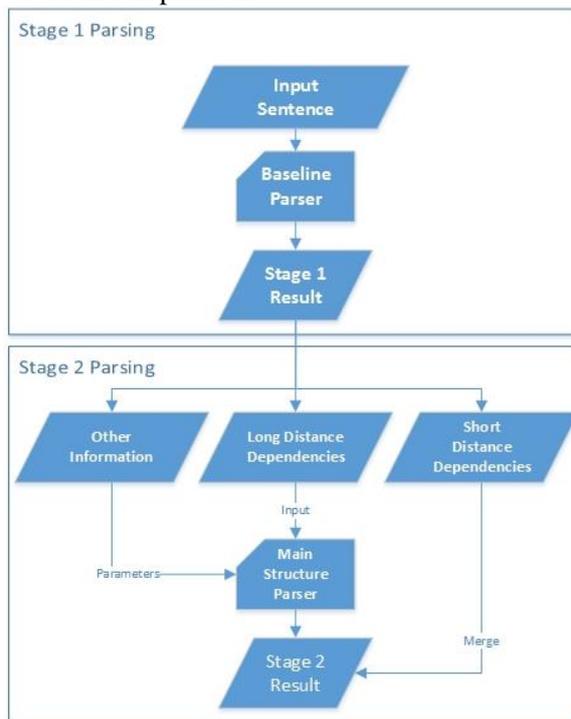


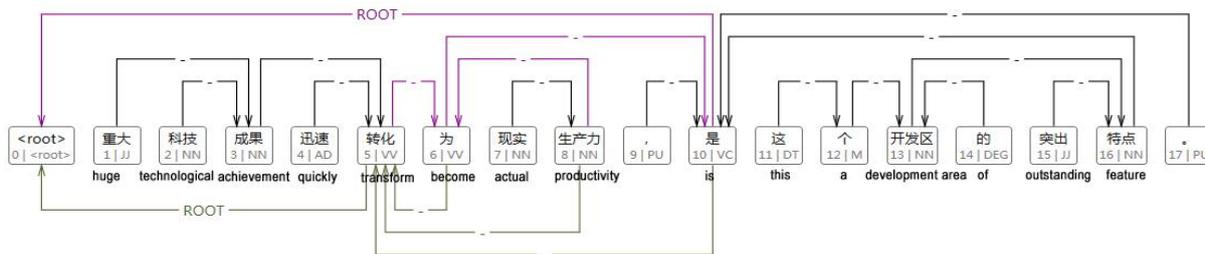
Figure 1: Framework of parser

### 3.2 Automatic Main Structure Words Extraction

Although main structure words are necessary for sentences, it is not easy to extract those words automatically. The differences between main structure words and non-main-structure words provide features to distinguish them.

Since it takes at least two components, namely head and its dependency, to form a dependency unit in a sentence, the parsing method also tries to find features of the main structure words from the two perspectives.

From the head perspective, the main structure words are normally the center constituent of its sub-sentence. The dependency relation between words could be regarded as a voting action. The more votes a word receives from its neighbor words, the more important the word is. Given the main structure words are usually the most important (important to the sentence structure) words, they tend to receive more votes from other words in its sub-sentence. Figure 2 (a) shows the process, the word 'Chengwei'(Becoming) and 'Touzi'(Investment) which have more incoming dependencies, are selected as main structure words from an example sentence (Figure 2 (a)).



Translation:

The fact that huge technological achievements can be quickly transformed into actual productivity, is the outstanding feature of this development area.

Figure 2 (a): An example sentence

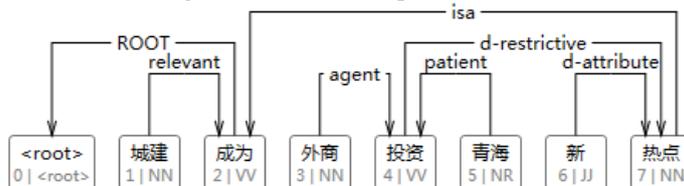


Figure 2 (b): word ‘ChengWei’ and ‘TouZi’ have more votes

From the dependency perspective, the main structure words are those words with long distance dependency. As main structure words are distributed among sub-sentences, the dependencies between main structure words are supposed to cross sub-sentences instead of being within a sub-sentence.

As previous discussion, three selection criteria that are significant in selecting main structure words are proposed: Crossing comma, the number of incoming dependencies and length of dependency. Based on the selection criteria, the following three methods for automatic main structure selection are attempted.

**Method 1: Crossing comma**

The idea of this method is to check whether a comma lines between the dependency head and its dependency. If the comma exists, both head and its dependency have been identified as main structure words.

The reason is that Chinese comma tends to represent weak stop between sub-sentences compared with that the English comma tends to represent weak stop between words or terms. In other words, the sub-sentences separated by commas are potential independent sentences. They just happen to be connected by commas because of the expression convention in Chinese. Given this, dependency that crosses a comma is of high chance to be between words that control structure of the sentence. Otherwise, the two sub-sentences are disconnected syntactically. This

situation is very common, especially in Chinese long sentences.

The process of the extraction is as follows.

Step 1: For each word  $w_i$  In a sentence  $S (w_0, w_1, w_2 \dots w_n)$ , find out all the incoming dependencies  $D_{ji}$  (dependency starting at word  $w_j$  and ending at word  $w_i, i \neq j$ ).

Step 2: For each incoming dependency  $D_{ji}$ , check if there is a word  $w_k (k \neq i, j)$  equal to the character comma (“,”).

Step 3: If there such a word  $w_k$ , the word  $w_i$  is selected as main structure word.

Figure 3 shows the result of the selecting method on the example sentence (Figure 2 (a)).

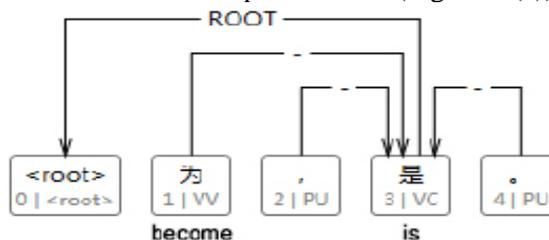


Figure 3: An example of Method 1

**Method 2: the number of incoming dependencies**

The idea behind this method is that the main structure words are always the center (root) of each sub-sentences, most other non-main structure words act as their modifier. Therefore, main structure words tend to have the most income dependencies from other words in each sub-sentence. The main structure words can be found

by counting the number of dependencies coming from other words.

Given that the number of sentences varies from sentence to sentence, the percentage instead of numbers are used as the measurement. The percentage is calculated within each sub-sentence rather than the whole sentence because the income dependency method does not work for long distance dependency. The process is as follows.

Step 1: For a sentence  $S (w_0, w_1, w_2 \dots w_n)$ , split it into sub-sentences by the character comma (“,”). The sub-sentences are:

$$S_{sub1}(w_0, w_1, w_2 \dots w_{i-1}),$$

$$S_{sub2}(w_{i+1}, w_{i+2}, w_{i+3} \dots w_{j-1})$$

...

$$S_{subn}(w_{n-k+1}, w_{n-k+2}, w_{n-k+3} \dots w_n)$$

Step 2: For each sub-sentence  $S_{subi}$ , calculate the number of words within the sub-sentence  $N_{subi}$ .

Step 3: For each word  $w_j$  in the sub-sentence  $S_{subi}$ , calculate the number of incoming dependencies  $N_{inj}$ .

Step 4: For each word  $w_j$  in the sub-sentence  $S_{subi}$ , calculate the percentage  $p(w_j) = \frac{N_{inj}}{N_{subi}}$ .

Step 5: Compare  $p(w_j)$  with pre-fixed threshold  $p$ , if  $p \leq p(w_j)$ , the word  $w_j$  is selected as main structure word.

**Method 3: Length of dependency**

This method uses the length of dependency as threshold to identify main structure words; this is enlightened by our observation that normally non-main structure words have short distance dependency because their dependencies are within the sub-sentence. Main structure words have dependencies with much longer lengths, so those words with length longer than normal situation are regarded as main structure words. The process is as follows.

Step 1: For each word  $w_i$  In a sentence  $S (w_0, w_1, w_2 \dots w_{n-1})$ , find out all the incoming dependencies  $D_{ji}$  (dependency starting at word  $w_j$  and ending at word  $w_i, i \neq j$ ).

Step 2: calculate the distance between  $i$  and  $j$   $Dis_{ij} = |i - j|$ .

Step 3: Normalize the Distance  $Dis_{ij}$  by dividing the whole length of sentence  $S$ , get length percentage  $P(w_i) = (Dis_{ij}/n) * 100\%$ .

Step 4: Compare  $p(w_i)$  with pre-fixed threshold  $p$ , if  $p \leq p(w_j)$ , the word  $w_j$  is selected as main structure word.

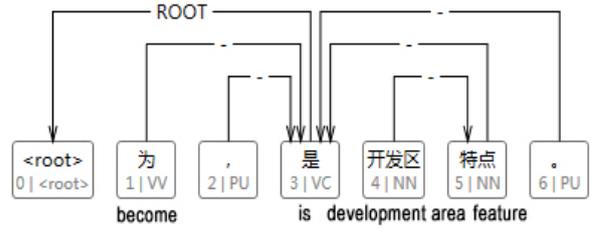


Figure 4: An example of Method 3

As there is a threshold parameter significantly affecting the performance in method 2 and method 3, a range of parameters examined over the whole testing set to find out the one with the best performance. Then our experiment was run over different length testing set to explore the best improvement it brings to baseline performance, which will, demonstrated in chapter 4.

**3.3 Dependency Voting**

According to previous discussion, the first stage parsing achieves high accuracy (around 90%) on short dependencies and low accuracy (around 30%) on long dependencies, while the second stage parsing has significantly better performance (around 40%, will discuss it in an experiment) on long distance dependencies. Some words (mostly main structure words) may have two parsing results, one from first stage parsing, and the other from second stage parsing. This paper uses a weighted voting scheme to decide what the final dependency for the words is. The weight of each parser comes from its accuracy on those specific parts. For example, baseline parser achieves around 84% accuracy in short distance dependency; when it predicates a short distance dependency, there are 84% possibility that the dependency is right. Each word receives two predicates from the two parsers; if the two predicates are the same, the result is the dependency. Otherwise, the parser with higher accuracy wins. That means the dependency is determined by the result from a more reliable parser. In this case, the two parser voting can be simplified as that baseline parser controls short distance dependencies parsing result, while the second stage parser controls the result of long distance dependency parsing.

**4 Result Analysis**

**4.1 Corpus**

We train and evaluate our parser on the dependency corpus called Chinese dependency Treebank 1.0 from Harbin Institute of Technology

(HIT). This corpus is available on the webpage of the conference of natural language processing and Chinese computing (NLPCC). Corpus follows the CoNLL2007 Standard, contains about 8,301 sentences in the training set, 534 sentences in the development set and 1,233 sentences in the test set. It has a much longer average sentence length than Penn Chinese Treebank (PCTB) (33 compare to 28 (Xue et al., 2005). For all experiments, we use the test set and report unlabeled attachment scores (UAS) for evaluation.

### 4.2 Baseline Parser

The baseline parser used in this paper is the Malt parser proposed by (Nivre, 2007). Based on the previous analysis, the state-of-art graph-based parser is slightly outperformed than transition-based parser while at the expense of surging scale of efficiency to  $O(n^2)$ . We aim to improve transition-based parser in order to maintain  $O(n)$  efficiency while improve accuracy as much as possible. In other words, there are other advanced parsers giving better performance than Malt parser in terms of accuracy, they are not selected because the processing speed in these parsers is sacrificed more or less. Compared with short sentence parsing, the importance of parsing efficiency (processing speed) in long sentence parsing is more significant. From this perspective, the Malt parser, providing  $O(n)$  efficiency with a little cost of accuracy is an ideal choice in this paper.

### 4.3 Experiments

Table 1 shows the test results of our parser on short dependencies. We include in the table results from the pure transition-based parser of (Zhang and Clark, 2008), the dynamic-programming arc-standard parser of (Huang and Sagae, 2010) and parsing with rich non-local features of (Zhang and Nirve, 2011) on Chinese. Our baseline parser and its extended methods are very close to its competing parsers in terms of the performance on short dependencies.

Table 2 shows the results of our parser on long dependencies (The dependencies between main structure words). Normally, the main structure words located in different sub-sentences of a long sentence. As a result, normal transition-based parsers cannot handle them well, which is the reason for the low scores overall. Our scores for this test set are the best reported so far and significantly better than the previous systems. In all our three methods, the best result is from the method 1, which improves the performance on

long dependencies by around 6.2% from baseline parser, while outperformance previous best system by 3.2%.

	UAS
Baseline	84.6%
Baseline +Method 1	84.3%
Baseline +Method 2	84.2%
Baseline +Method 3	84.4%
Zhang and Clark 2008	84.3%
Huang and Sagae 2010	85.2%
Zhang, Y., & Nivre 2011	86.0%

Table 1: Performance on short dependencies

	UAS
Baseline	32.1%
Baseline +Method 1	38.3%
Baseline +Method 2	36.6%
Baseline +Method 3	37.0%
Zhang and Clark 2008	33.3%
Huang and Sagae 2010	32.9%
Zhang, Y, & Nivre 2011	35.1%

Table 2: Performance on long dependencies

### 4.4 Parameter Optimization

In the three methods, the performance of method 2 and method 3 largely affected by threshold parameters, Table 3 shows the relationship between the threshold and accuracy, the best performance of method 2 achieved when the threshold set to be 35%.

Threshold	Accuracy	Threshold	Accuracy
1.00	29.5%	0.50	34.0%
0.95	29.6%	0.45	34.3%
0.90	29.8%	0.40	35.4%
0.85	29.8%	<b>0.35</b>	<b>36.1%</b>
0.80	30.0%	0.30	35.3%
0.75	30.3%	0.25	33.3%
0.70	30.9%	0.20	31.8%
0.65	30.7%	0.15	30.7%
0.60	32.0%	0.10	28.2%
0.55	33.1%	0.05	26.4%

Table 3: Performance of Method 2 with Different Threshold

The method 2 achieves the best performance when there are not enough words chosen as main structure words. For example, the 1.00 means that one word has to get all dependencies from other words, within its sub-sentence to be selected as the main structure word, and this is almost impossible. Lower the threshold also deteriorates

the performance because that means more words are chosen as main structure words. For example, the 0.00 means all words are main structure words, and they are all parsed by stage 2 parsers which training in a main structure corpus.

Threshold	Accuracy	Threshold	Accuracy
0	11.7%	26	33.0%
2	21.1%	28	32.6%
4	27.1%	30	32.6%
6	30.9%	32	32.4%
8	32.9%	34	32.3%
10	32.6%	36	32.3%
12	33.5%	38	32.2%
14	33.6%	40	32.2%
<b>16</b>	<b>33.8%</b>	42	32.1%
18	33.4%	44	32.1%
20	33.3%	46	32.0%
24	33.3%	48	32.0%

Table 4: Performance of Method 3 with Different Threshold

Table 4 shows the performances of the main structure parser with method 3 with different parameters. The method achieves the best performance when the parameter is set to 16, the performance curve before and after this point experience a comparable decrease like method 2.

Length	Baseline	M1	M2	M3
40-50	33.2%	<b>40.0%</b>	38.2%	33.4%
50-60	32.4%	36.4%	35.5%	32.0%
60-70	32.2%	36.9%	<b>37.2%</b>	33.2%
70-80	28.8%	34.5%	35.2%	31.8%
80-90	22.0%	30.2%	29.1%	26.8%
90-100	23.5%	30.5%	26.5%	28.1%
100-110	26.1%	29.1%	26.7%	<b>32.0%</b>
110-120	25.7%	29.2%	28.4%	31.1%
120-130	19.0%	22.5%	21.7%	24.4%
130-140	21.1%	32.1%	22.2%	27.8%
140-150	20.0%	21.9%	27.1%	24.3%

Table 5: Method comparison on each sub-set

As can be seen from section 4.4, method 1 outperformed the other two methods in both UAS and ULAS, Table 5 shows that the method 3 achieves better performance than method 1 when the length of sentence is longer than 100, while it is worse than method 2 in sentences with length less than 80.

## 5 Conclusion

This research proposes a two-stage parsing method called main structure parsing, used to improve the parsing performance for Chinese long sentence. The performance of normal dependency parser decreases on long sentences because of the long distance dependencies between main structure words. The main structure parsing method alleviates long dependency problem by selecting out the main structure words and parse them with a specially trained parser. Three different methods regarding selecting those main structure words are compared and tested in the thesis. The best method achieves a 6.0 % improvement on long dependency than baseline parser and 3.2% improvement than the previous best mode.

## Reference

Jones, B. (1996). What’s the point? A (computational) theory of punctuation.

Bohnet, B., & Kuhn, J. (2012, April). The best of both worlds: a graph-based completion model for transition-based parsers. In Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics (pp. 77-87). Association for Computational Linguistics.

Candito, M., & Seddah, D. (2012, November). Effectively long-distance dependencies in French: annotation and parsing evaluation. In TLT 11-The 11th International Workshop on Treebanks and Linguistic Theories.

Chang, C. C., & Lin, C. J. (2011). LIBSVM: A Library for Support Vector Machines. *ACM Transactions on Intelligent Systems and Technology*, 2: 27: 1–27: 27, 2011. Software available at <http://www.csie.ntu.edu.tw/~cjlin/libsvm>.

Zhao, Y., Che, W., Guo, H., Qin, B., Su, Z., & Liu, T. (2014). Sentence compression for target-polarity word collocation extraction. In Proceedings of COLING (pp. 1360-1369).

Che, W., Spitkovsky, V. I., & Liu, T. (2012, July). A comparison of chinese parsers for stanford dependencies. In Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Short Papers-Volume 2 (pp. 11-16). Association for Computational Linguistics.

Ferreira, F., Engelhardt, P. E., & Jones, M. W. (2009). Good enough language processing: A satisficing approach. In Proceedings of the 31st Annual conference of the Cognitive Science Society. Austin: Cognitive Science Society.

- Yamada, H., & Matsumoto, Y. (2003, April). Statistical dependency analysis with support vector machines. In *Proceedings of IWPT (Vol. 3, pp. 195-206)*.
- Lai, B. Y. T., & Huang, C. (1994). Dependency grammar and the parsing of Chinese sentences. arXiv preprint [cmp-lg/9412001](https://arxiv.org/abs/cmp-lg/9412001). Lai, T. B., Huang, C., Zhou, M., Miao, J., Siu, T. K., 2001. Span-based statistical dependency parsing of Chinese. In: *NLPRS*. pp. 677–684.
- Li, X., Zong, C., & Hu, R. (2005). A Hierarchical Parsing Approach with Punctuation Processing for Long Sentence Sentences. In *In Proceedings of the Second International Joint Conference on Natural Language Processing: Companion Volume including Posters/Demos and Tutorial Abstracts*.
- Li, Z., Che, W., & Liu, T. (2010, December). Improving dependency parsing using punctuation. In *Asian Language Processing (IALP), 2010 International Conference on (pp. 53-56)*. IEEE.
- Xun Jin, M., Kim, M. Y., Kim, D., & Lee, J. H. (2004). Segmentation of Chinese long sentences using commas. In *Proceedings of SIGHAN (pp. 1-8)*.
- Covington, M. A. (2001). A fundamental algorithm for dependency parsing. In *Proceedings of the 39th annual ACM southeast conference (pp. 95-102)*.
- Nivre, J., & McDonald, R. T. (2008, June). Integrating Graph-Based and Transition-Based Dependency Parsers. In *ACL (pp. 950-958)*.
- Nivre, J., Hall, J., & Nilsson, J. (2006, May). Malt-parser: A data-driven parser-generator for dependency parsing. In *Proceedings of LREC (Vol. 6, pp. 2216-2219)*.
- Nilsson, J., Riedel, S., & Yuret, D. (2007, June). The CoNLL 2007 shared task on dependency parsing. In *Proceedings of the CoNLL shared task session of EMNLP-CoNLL (pp. 915-932)*.
- Nivre, J., & McDonald, R. T. (2008, June). Integrating Graph-Based and Transition-Based Dependency Parsers. In *ACL (pp. 950-958)*.
- MAO, Q., LIAN, L. X., ZHOU, W. C., & YUAN, C. F. (2007). Chinese syntactic parsing algorithm based on segmentation of punctuation. *Journal of Chinese Information Processing*, 21(2), 3.
- Sagae, K and Lavie, A. 2006a. Parser combination by reparsing. In *Proc. HLT/NAACL*, pages 129–132, New York City, USA, June.
- Sagae, K., & Lavie, A. (2006, June). Parser combination by reparsing. In *Proceedings of the Human Language Technology Conference of the NAACL, Companion Volume: Short Papers (pp. 129-132)*. Association for Computational Linguistics.
- Wang, W. Y., Kong, L., Mazaitis, K., & Cohen, W. W. (2014). Dependency Parsing for Weibo: An Efficient Probabilistic Logic Programming Approach. *Association for Computational Linguistics*.
- Xue, N., Xia, F., Chiou, F. D., & Palmer, M. (2005). The Penn Chinese TreeBank: Phrase structure annotation of a large corpus. *Natural language engineering*, 11(02), 207-238.
- Zhou, M. (2000, October). A block-based robust dependency parser for unrestricted Chinese text. In *Proceedings of the second workshop on Chinese language processing: held in conjunction with the 38th Annual Meeting of the Association for Computational Linguistics-Volume 12 (pp. 78-84)*. Association for Computational Linguistics.
- Zhang, Y., & Nivre, J. (2011, June). Transition-based dependency parsing with rich non-local features. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: short papers-Volume 2 (pp. 188-193)*. Association for Computational Linguistics.

# A Light Rule-based Approach to English Subject-Verb Agreement Errors on the Third Person Singular Forms

Yuzhu Wang<sup>1,2</sup>, Hai Zhao<sup>1,2</sup> \* † and Dan Shi<sup>3</sup>

<sup>1</sup>Center for Brain-Like Computing and Machine Intelligence,  
Department of Computer Science and Engineering,  
Shanghai Jiao Tong University, Shanghai, 200240, China

<sup>2</sup>Key Laboratory of Shanghai Education Commission for Intelligent Interaction  
and Cognitive Engineering, Shanghai Jiao Tong University, Shanghai, 200240, China

<sup>3</sup>LangYing NLP Research Institute, Shanghai LangYing Education Technology Co., Ltd.  
hfut0830@sjtu.edu.cn, zhaohai@cs.sjtu.edu.cn, sweet@lyced.com

## Abstract

Verb errors are one of the most common grammar errors made by non-native writers of English. This work especially focus on an important type of verb usage errors, subject-verb agreement for the third person singular forms, which has a high proportion in errors made by non-native English learners. Existing work has not given a satisfied solution for this task, in which those using supervised learning method usually fail to output good enough performance, and rule-based methods depend on advanced linguistic resources such as syntactic parsers. In this paper, we propose a rule-based method to detect and correct the concerned errors. The proposed method relies on a series of rules to automatically locate subject and predicate in four types of sentences. The evaluation shows that the proposed method gives state-of-the-art performance with quite limited linguistic resources.

## 1 Introduction

With the increasing number of people all over the world who study English as second language (ESL), grammatical errors in writing often occur due to cultural diversity, language habits, and education background. There has been a substantial and increasing need of using computational techniques to improve the writing ability for second language learners. In addition, such techniques and tools may help find latent writing errors in official documents as well. To meet the urgent need from ESL, a lot of works on natural language processing focus on the task of grammatical error detection and correction. Formally, it is a task of automatically detecting and correcting erroneous word usage and ill-formed grammatical constructions in text (Dahlmeier et al., 2012).

It is not a brand new task in natural language processing. However, it has been a challenging task for several reasons. First, many of these errors are context-sensitive so that errors cannot be detected and then corrected in an isolated way. Second, the relative frequency of errors is quite low: for a given type of mistake, an ESL writer will typically go wrong in only a small proportion of relevant language structures. For example, incorrect determiner usages usually occur in 5% to 10% of noun phrases in various annotated ESL corpora (Rozovskaya and Roth, 2011). Third, an ESL writer may make multiple mistakes in a single sentence, so that continuous errors are entangled, which let specific error locating and correction become more difficult.

\* Correspondence author

† This work was partially supported by the National Natural Science Foundation of China (No. 61170114, and No. 61272248), the National Basic Research Program of China (No. 2013CB329401), the Science and Technology Commission of Shanghai Municipality (No. 13511500200), the European Union Seventh Framework Program (No. 247619), the Cai Yuanpei Program (CSC fund 201304490199 and 201304490171), and the art and science interdisciplinary funds of Shanghai Jiao Tong University, No. 14X190040031, and the Key Project of National Society Science Foundation of China, No. 15-ZDA041.

In recent decades, existing studies on this task have focused on errors in two typical word categories, article and preposition (Han et al., 2006; Felice and Pulman, 2008; Dahlmeier and Ng, 2011). However verb errors occur as often as article and preposition errors at least, though there are few works on verb related errors. Two reasons are speculated for why it is difficult to process verb mistakes. First, compared with articles and prepositions, verbs are more difficult to identify in text, as they can often be confused with other parts of speech (POS), and in fact many existing processing tools are known to make more errors on noisy ESL data (Nagata et al., 2011). Second, verbs are more complicated linguistically. For an English verb, it has five forms of inflections (see Table 1). Different forms imply different types of errors, even, one type of verb form may lead to multiple types of errors.

Form	Example
base(bare)	<i>Speak</i>
base(infinitive)	<i>to speak</i>
third person singular	<i>speaks</i>
past	<i>spoke</i>
-ing participle	<i>speaking</i>
-ed participle	<i>spoken</i>

Table 1: Five forms of inflections of English verbs (Quirk et al., 1985), illustrated with the verb “*speak*”. The base form is also used to construct the infinitive with “*to*”.

China is a leading market for ESL. According to a rough statistics on essays written by Chinese students, verb related errors have given a percent as high as 15.6% among all grammatical errors, in which subject-verb agreement errors on the third person singular form cover 21.8%. Existing works paid little attention on such type of errors, or report unsatisfied performance (Rozovskaya et al., 2013). That is to say, errors made by Chinese students have a quite different type distribution from those by native English students, while existing computational approach cannot well meet the urgent requirement on grammatical error detection and correction. Furthermore, the previous approaches focus on machine learning that always needs a large scale of annotated data set available. However, being a machine learning task, grammatical error detection

and correction is very difficult to receive satisfied performance as errors being negative samples has too low a portion in the entire text for learning (on average, 20 sentences can hold one error).

In this paper, to alleviate the drawbacks of existing work, we propose a full rule-based method to handle this sort of specific errors, without any requirement on annotated data. The rule model is built on the English grammar. As we avoid using high-level and time consuming support tools, typically, parser, only two lexicons and a part-of-speech (POS) tagger <sup>1</sup> (Toutanova et al., 2003) is adopted to provide necessary word category information. This makes our system can work with least linguistic resource compared to existing rule-based work.

The rest of this paper is organized as follows: Section 2 discusses a few related work. Section 3 gives detailed introduction about the proposed rule-based method. The experimental results will be presented and analyzed in Section 4, and the last section concludes this paper.

## 2 Related Work

Over the past few decades, there are many methods proposed for grammatical error detection and correction. Most of the efforts so far had been focused on article and preposition usage errors, as these were some of the most common mistakes among non-native English speakers (Dalgish, 1985; Leacock et al., 2010). These works were generally regarded as multiclass classification tasks (Izumi et al., 2003; Han et al., 2006; Felice and Pulman, 2008; Gamon et al., 2008; Tetreault et al., 2010; Rozovskaya and Roth, 2010b; Rozovskaya and Roth, 2011; Dahlmeier and Ng, 2011).

As for main techniques for the task, most methods can fall into two basic categories, machine learning based and rule-based. The use of machine learning methods to tackle this problem had shown a promising performance for specific error types. These methods were normally created based on a large corpus of well-formed native English texts (Tetreault and Chodorow, 2008; Tetreault et al., 2010) or annotated non-native data (Gamon, 2010;

<sup>1</sup>This POS tagger outputs a POS tag set as the same defined by Penn Treebank.

Han et al., 2010). Additionally, both generative and discriminative classifiers were widely used. Among them, Maximum Entropy (Rozovskaya and Roth, 2011; Sakaguchi et al., 2012; Quan et al., 2012) obtained a good result for preposition and article correction using a large feature set. Naive Bayes was also applied to recognize or correct the errors in speech or texts (Lynch et al., 2012). In addition, grammar rules and probabilistic language model were used as a simple but effective assistant for correction of spelling (Kantrowitz, 2003) and grammatical errors (Dahlmeier et al., 2012; Lynch et al., 2012; Quan et al., 2012; Rozovskaya et al., 2012).

As for rule-based method, (Rozovskaya et al., 2014) proposed a linguistically-motivated approach to verb error correction that made use of the notion of verb finiteness to identify triggers and types of mistakes, before using a statistical machine learning approach to correct these mistakes. In their approach, the knowledge of which mistakes should be corrected or of the mistake type was not required. But their model got a low recall.

Recently, researchers also made an attempt to integrate different methods. (Rozovskaya et al., 2013) presented a system that combined a set of statistical models, where each model specialized in correction one of the five type errors which were article, preposition, noun number, verb form and subject-verb agreement. Their article and preposition modules built on the elements of the systems described in (Rozovskaya and Roth, 2011).

(Gamon et al., 2009) mentioned a model for learning gerund/infinitive confusions and auxiliary verb presence/choice. (Lee and Seneff, 2008) proposed an approach based on pattern matching on trees combined with word  $n$ -gram counts for correcting agreement misuse and some types of verb form errors. However, they excluded tense mistakes. (Tajirei et al., 2012) considered only tense mistakes. In the above studies, it was assumed that the type of mistake that needs to be corrected is known, and irrelevant verb errors were excluded (Tajirei et al., 2012) addressed only tense mistakes and excluded from the evaluation other kinds of verb errors.

### 3 Our Approach

Our approach requires two lexicons and a POS tagger as the basic linguistic resource to perform the task. As for the POS tagger, we use the POS tag set defined by Penn treebank. It has 36 POS tags, and each has a specific syntactic or even semantic role, which is shown in Table 2. The detailed roles of these POS tags will give basic criterion to locate subject and its predicate in a sentence.

As for lexicons, it is used to determine if a verb is in root form or not. To judge whether a verb has an agreement error, we build two dictionaries. One consists of 2,677 original verbs which are extracted from Oxford Advanced Learner's Dictionary (Hornby et al., 2009). The other contains all 2,677 verbs in the third person singular form. We find that there is not a word which exists in both dictionaries, so we can decide whether a verb is in the root form or in the third person form by checking the verb in which dictionary. Then the remaining job is to locate the subject and its predicate. Linguistically, subject and predicate can be either syntactic or semantic. The subject in syntax (grammar) and semantics may be the same in a few cases, but different in the others. For an interrogative sentence such as "*who are you?*", "*who*" is the true subject in grammar, however, what we always need is the semantic or nominal subject "*you*", so that we can check the agreement between "*you*" and its predicate "*are*". Throughout the entire paper, our rules and processing always take subject and its predicates as the semantic or nominal ones.

According to the different relative locations of subject and its predicate in sentences, we put all sentences into four categories, declarative, interrogative, subordinate and "*there be*" sentences. These sentence categories will be effectively determined through limited number of rules on specific punctuations and marker words. For declarative sentences, subject is before its predicate. For interrogative sentences, there is no fixed location relation between subjects and its predicates. For "*there be*" sentences, the nominal subject is after the predicate "*be*".

POS Tag	Description
<i>CC</i>	Coordinating conjunction
<i>CD</i>	Cardinal number
<i>DT</i>	Determiner
<i>EX</i>	Existential <i>there</i>
<i>FW</i>	Foreign word
<i>IN</i>	Preposition or subordinating conjunction
<i>JJ</i>	Adjective
<i>JJR</i>	Adjective, comparative
<i>JJS</i>	Adjective, superlative
<i>LS</i>	List item marker
<i>MD</i>	Modal
<i>NN</i>	Noun, singular or mass
<i>NNS</i>	Noun, plural
<i>NNP</i>	Proper noun, singular
<i>NNPS</i>	Proper noun, plural
<i>PDT</i>	Predeterminer
<i>POS</i>	Possessive ending
<i>PRP</i>	Personal pronoun
<i>PRP\$</i>	Possessive pronoun
<i>RB</i>	Adverb
<i>RBR</i>	Adverb, comparative
<i>RBS</i>	Adverb, superlative
<i>RP</i>	Particle
<i>SYM</i>	Symbol
<i>TO</i>	<i>to</i>
<i>UH</i>	Interjection
<i>VB</i>	Verb, base form
<i>VBD</i>	Verb, past tense
<i>VBG</i>	Verb, gerund or present participle
<i>VBN</i>	Verb, past participle
<i>VBP</i>	Verb, non-3rd person singular present
<i>VBZ</i>	Verb, 3rd person singular present
<i>WDT</i>	Wh-determiner
<i>WP</i>	Wh-pronoun
<i>WP\$</i>	Possessive wh-pronoun
<i>WRB</i>	Wh-adverb

Table 2: Penn Treebank POS tag set

### 3.1 Declarative Sentences

For declarative sentences, predicate can be easily determined by searching for the first verb from the beginning of the sentence. Because most of the subjects are either nouns or pronouns, we continue to scan the sentence from beginning to the position of the predicate to confirm the subject. Except the case that the subject is “*I*” whose predicate must be “*am*”, all the subjects can be divided into the third person singular and the non-third person singular. For noun, we regard the words with POS tag “*NN*” as the third person singular and the words with POS tag “*NNS*” as the non-third person singular. For pronoun, we collect two lists (see Table 3) to distinguish whether the subject is the third person singular. Note that a person name can also be subject and we regard the name as the third person singular. We can utilize the POS tag “*NNP*” and “*NNPS*” to locate a person name. For this case, we continue to scan the sentence from the position of subject to find a verb.

Third Person Singular	Non Third Person Singular
<i>He_PRP</i>	<i>You_PRP</i>
<i>he_PRP</i>	<i>you_PRP</i>
<i>She_PRP</i>	<i>We_PRP</i>
<i>she_PRP</i>	<i>we_PRP</i>
<i>It_PRP</i>	<i>They_PRP</i>
<i>it_PRP</i>	<i>they_PRP</i>
<i>That_DT</i>	<i>These_DT</i>
<i>that_WDT</i>	<i>these_DT</i>
<i>This_DT</i>	<i>Those_DT</i>
<i>this_DT</i>	<i>those_DT</i>
<i>That_WDT</i>	<i>us_PRP</i>

Table 3: Pronouns of the third person and none third person (with POS tags)

With the above processes, we will still receive a wrong result for specific sentences with compound subject. For example, “*Tom and Jack come from America .*”. So we need to add a rule to process these compound subjects. The desired subject can be determined by checking if it is after a word and POS tag combination, “*and\_CC*”, which means that the word is “*and*” as a conjunction for the case that the subject is determined to be third person.

Although we can deal with most of the simple

sentences so far, there are also many sentences which can not be process according to these rules.

Firstly, for the sentences which have a modal verb before the predicate, the wanted verb must be in the original form no matter the subject is third person. We can identify this case by searching POS tag “MD” between the subject and the verb.

Secondly, there are often many compound sentences in statement. For example,

1. “He likes apple but she like orange .”
2. “She will name him whatever she want to .”
3. “I love her because she give me life .”
4. “As we all know , human can not live without water .”

For these cases, we divide the sentences into two parts and handle the rest part as declarative sentence recursively. For sentences like example 1-3, we build a list which consists of the words called *separate word* (see Table 4). We split the sentences by means of finding the *separate word*. For the sentences like example 4, the comma mark is used as the splitting boundary. We can utilize the words called *guided word* (see Table 5) to identify this type of sentences.

<i>and_CC</i>	<i>but_CC</i>
<i>so_RB</i>	<i>or_CC</i>
<i>because_IN</i>	<i>nor_CC</i>
<i>whatever_WDT</i>	<i>whatever_WPT</i>
<i>whether_IN</i>	<i>what_WP</i>
<i>why_WRB</i>	<i>where_WRB</i>
<i>when_WRB</i>	<i>how_WRB</i>
<i>whose_WPS</i>	<i>that_IN</i>
<i>before_IN</i>	<i>if_IN</i>
<i>wherever_WPT</i>	

Table 4: The *separate words* (with POS tags)

<i>As_IN</i>	<i>If_IN</i>
<i>Although_IN</i>	<i>When_WRB</i>
<i>So_RB far_RB as_IN</i>	

Table 5: The *guided words* (with POS tags)

However, for sentences that were led by a prepositional phrase, the rules proposed above can not correctly deal with. Here are two examples:

1. “In my view, they are right .”

2. “In the morning , the dogs are running on the road .”

We will regard the “view” and “morning” as subject according to the existing rules. But the true subjects are “they” and “dogs”. So if there is “In\_IN” before the noun, we will abandon the noun and regard the rest of the sentence as a new sentence for processing.

### 3.2 Interrogative Sentences

In English grammar, questions mainly contain four categories. They are general question, alternative question, special question and tag question. Here are four examples:

1. “Are you student ?”
2. “Can you speak Chinese or English ?”
3. “Who are you ?”
4. “They work hard , don’t they ?”

As in general predicate is before subject in most interrogative sentences, we scan the sentence from the beginning and regard the first verb as the predicate according to POS tag “VB”. Then we continue to scan the sentence until the subject is found. The rules are the same as those proposed for declarative sentences.

Note that a tag question consists of two parts, a declarative sentence and a general question in abbreviation form. So we must divide the disjunctive question into two parts and process the first part as declarative sentence. Note that the fourth symbol from the end is a comma in all tag questions. We will make a full use of this mark to effectively divide a tag question.

There are also a few sentences that deserve our attention. For instance,

1. “Whose jeans are they ?”
2. “How many boys are there ?”.

We can find that subject is in front of predicate in these sentences, so we can simply regard these sentence as declarative sentences. These types of sentences can be found by checking if they start from words like “Whose\_JJ”, “How\_WRB many\_JJ” and “How\_WRB much\_RB”.

### 3.3 Subordinate Clause

So far, we have considered most of simple sentences. But there are many compound sentences with subordinate clause in real expression. We

furthermore divide the sentences with subordinate clause into five categories. Here are five examples:

1. “*The girl who is speaking now comes from Japan .*”
2. “*He gives me a gift which is very beautiful .*”
3. “*What she wants is a lovely doll .*”
4. “*The club will give whoever wins the competition a prize .*”
5. “*She will give him whatever he wants to .*”

For the first and second categories, we need pay attention to the conjunctions “*who*”, “*which*” and “*that*”. But the positions of the conjunctions are different in first and second categories. For the sentence like example 1, we check whether there is a conjunction between subject and predicate. If we find the conjunctions, we regard both the first and the second verbs as the predicate with the same subject.

For the second category, we check whether there is a conjunction after the predicate. If the conjunction is found, we will scan the sentence from the position of the conjunction to the position of predicate to find the subject of subordinate clause. The rules and treatments used to find the subject are the same as those proposed for declarative sentence. At last we scan the sentence from the position of conjunction to the end to find the predicate of subordinate clause.

For a sentence as example 3, we check whether the sentence begins with “*What\_WP*” or “*Whether\_IN*”. If it is, we regard the second verb as the predicate of the subordinate clause and consider the subject of the subordinate clause as the third person. If we find “*whoever\_WP*” after the verb in a sentence, we will scan the sentence from the position of “*whoever\_WP*” to the end to find the second predicate and consider its subject as the third person.

For the last category, we divide the sentence into two parts by locating the word “*whatever\_WP*” and handle both parts as declarative sentences.

### 3.4 “*There be*” Sentences

The semantic subject of “*there be*” sentence is the first noun right after the verb “*be*”. Note that sentences like “*Here is five questions to be answered.*” also can be regard as “*there be*” sentences. All these types of sentences can be identified by searching the leading words

“*There\_EX*” and “*Here\_RB*”.

### 3.5 Additional Rules

Although most of the sentences can be processed by the proposed rules now, there are still some very special cases that can not be handled. Moreover, the outputs of POS tagger are not exact completely. So we give a few additional rules to strengthen the model.

Firstly, the words like “*Chinese*” are third person when they mean a language, otherwise, they are not. We call these words *language words*. We observe that when the *language word* means *language*, there is always a word “*language*” in the sentence. So we check whether there is “*language*” in the sentence that contains a *language word*. If we find “*language*”, we will compulsively modify the corresponding word with the an updated POS tag “*NN*”. Otherwise, we change the word with the an updated POS tag “*NNS*”. There is also a situation that the subject is a gerund sometimes. We know that the gerund can not be a predicate by itself. So we change all the gerunds with the POS tag “*NN*”. Table 6 shows additional rules to fortify the model.

### 3.6 Correction

Because there is not a word in both original form and third person form and one verb only has one third person form, we build a mapping dictionary to map a word from its root form to the third person singular form. Each word that is detected as error can be restored by searching this mapping dictionary.

## 4 Result

We select 300 sentences with agreement errors and 3,000 correct sentences from essays written by Chinese students as the test data. This data set is provided by Shanghai LangYing Education Technology Co., Ltd.. The results are evaluated by the metrics, precision  $P$ , recall  $R$  of error detection and correction, and their harmonic average  $F1$  score (Table 7). As Lee model (Lee and Seneff, 2008) can process subject-verb agreement errors well, we compare their results with ours on the same test data set<sup>2</sup>.

<sup>2</sup>As (Lee and Seneff, 2008) do not release their data set and system implementation, we have accurately re-implement their system to make this comparison.

The case need to be handled	The rules
If there is “Not only”.	Abandon all the words before “also”
If there is “I think”.	Check whether “I think” is wrong then abandon “I think”.
If there is “percent of”.	Abandon “percent of”.
If there is “a lot of ”.	Abandon “a lot of ”.
If there is “a number of”.	Abandon “a number of”.

Table 6: Additional rules

The comparison in Table 7 shows that our model outperforms Lee model by 6.7% in terms of F1 score. In addition, the results of Lee model were achieved by adopting advanced parse tree, while we use no more than POS tags.

We also show the result of Rozovskaya model (Rozovskaya et al., 2014) and UIUC model (Rozovskaya et al., 2013) (see Table 8 and 9). Our model is significantly better than theirs for subject-verb agreement errors though their model can deal with various types of errors. However, it is worth noting that their test data sets are different for all existing works and ours. Therefore, we compare their results only for reference.

### 5 Conclusion

Verb errors are commonly made by ESL writers but difficult to process. Subject-verb agreement errors on the third person singular form cover 21.8% of

Model		P	R	F1
Our Model	Identification	85.0	81.7	83.3
	Correction	85.0	81.7	83.3
Lee Model	Identification	82.3	71.6	76.6
	Correction	82.3	71.6	76.6

Table 7: Results

Models	P	R	F1
<i>Scores on the original annotations</i>			
Articles	48	11	18
+Prepositions	48	12	19
+Noun number	48	21	29
+Subject-verb agr	48	22	30
+Verb form(All)	46	23	31
<i>Scores based on the revised annotations</i>			
All	62	32	42

Table 9: Results of the UIUC model

the verb errors according to statistics from a typical ESL group. Previous works paid little attention on such type of errors, and report unsatisfied performance. Using quite limited linguistic resources, we develop a rule-based approach that gives state-of-the-art performance on detecting and correcting the subject-verb agreement errors.

### References

A S Hornby, Sally Wehmeier and Michael Ashby. 2009. *Oxford Advanced Learner’s Dictionary* . Oxford University Press, Oxford, England.

Alla Rozovskaya and Dan Roth. 2010b. *Training paradigms for correcting errors in grammar and usage*. In Proceedings of the 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics, pp. 154-162.

Alla Rozovskaya and Dan Roth. 2011. *Annotating ESL errors: Challenges and rewards*. In Proceedings of the NAACL Workshop on Innovative Use of NLP for Building Educational Applications, pp. 28-36

Alla Rozovskaya, Dan Roth and Srikumar Vivek. 2014. *Correcting Grammatical Verb Errors* . In Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics, pp. 358-367

Alla Rozovskaya, Kaiwei Chang, Mark Sammons and Dan Roth. 2013. *The University of Illinois System in the CoNLL-2013 Shared Task* . In Proceedings of the Seventeenth Conference on Computational Natural Language Learning: Shared Task, pp. 13-19

Alla Rozovskaya, Mark Sammons and Dan Roth. 2012. *The UI system in the HOO 2012 shared task on error correction* . In Proceedings of the Seventh Workshop on Building Educational Applications Using NLP, pp. 272C280.

Chang-Ning Huang and Hai Zhao. 2006. *Which Is Essential for Chinese Word Segmentation: Character versus Word* . In Proceedings of the 20th Pacific Asia Conference on Language, Information and Computation (PACLIC-20), pp. 1-12

Error type	Correction			Identification		
	P	R	F1	P	R	F1
Agreement	90.62	9.70	17.52	90.62	9.70	17.52
Tense	60.51	7.47	13.31	86.63	10.70	19.06
Form	81.83	16.34	27.24	83.47	16.67	27.79
Total	71.94	10.24	17.94	85.81	12.22	21.20

Table 8: Results of Rozovskaya model

- Claudia Leacock, Martin Chodorow, Michael Gamon and Joel Tetreault. 2010. *Automated Grammatical Error Detection for Language Learners*. Morgan and Claypool Publishers
- Daniel Dahlmeier and Hwee Tou Ng. 2011. *Grammatical Error Correction with Alternating Structure Optimization*. In Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, pp. 915-923
- Daniel Dahlmeier, Hwee Tou Ng and Eric Jun Feng Ng. 2012. *NUS at the HOO 2012 Shared Task*. In Proceedings of the Seventh Workshop on Building Educational Applications Using NLP, pp. 216-224
- Emi Izumi, Kiyotaka Uchimoto, Toyomi Saiga, Thepchai Supnithi and Hitoshi Isahara. 2003. *Automated Grammatical Error Detection for Language Learners*. In Proceedings of 41st Annual Meeting of the Association for Computational Linguistics, pp. 145-148
- G. Dalgish. 1985. *Computer-assisted ESL research*. CALICO Journal, 2(2)
- Gerard Lynch, Erwan Moreau and Carl Vogel. 2012. *A Naive Bayes classifier for automatic correction of preposition and determiner errors in ESL text*. In Proceedings of the Seventh Workshop on Building Educational Applications Using NLP, pp. 257-262.
- Hai Zhao and Chunyu Kit. 2007. *Incorporating Global Information into Supervised Learning for Chinese Word Segmentation*. In Proceedings of the 10th Conference of the Pacific Association for Computational Linguistics (PACLING-2007), pp. 66-74
- Hai Zhao, Chang-Ning Huang and Mu Li. 2006. *An Improved Chinese Word Segmentation System with Conditional Random Field*. In Proceedings of the Fifth SIGHAN Workshop on Chinese Language Processing (SIGHAN-5), pp. 162-165
- Hai Zhao, Wenliang Chen, Jun'ichi Kazama, Kiyotaka Uchimoto, and Kentaro Torisawa. 2009. *Multilingual Dependency Learning: A Huge Feature Engineering Method to Semantic Dependency Parsing*. In Proceedings of Thirteenth Conference on Computational Natural Language Learning, pp. 55-60
- Hai Zhao, Xiaotian Zhang, and Chunyu Kit. 2013. *Integrative Semantic Dependency Parsing via Efficient Large-scale Feature Selection*. Journal of Artificial Intelligence Research, Volume 46:203-233
- Hai Zhao, Yan Song, Chunyu Kit, and Guodong Zhou. 2009. *Cross Language Dependency Parsing using a Bilingual Lexicon*. Joint conference of the 47th Annual Meeting of the Association for Computational Linguistics and the 4th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing, pp. 55-63
- Hai Zhao. 2009. *Character-Level Dependencies in Chinese: Usefulness and Learning*. In Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics, pp. 879-887
- Jian Zhang, Hai Zhao, Liqing Zhang, and Baoliang Lu. 2011. *An Empirical Comparative Study on Two Large-Scale Hierarchical Text Classification Approaches*. International Journal Computer Processing of Oriental Language, pp. 309-326
- Jingyi Zhang and Hai Zhao. 2013. *Improving Function Word Alignment with Frequency and Syntactic Information*. In Proceedings of International Joint Conference on Artificial Intelligence-2013, pp. 2211-2217
- Joel R. Tetreault and Martin Chodorow. 2008. *The ups and downs of preposition error detection in ESL writing*. In Proceedings of the 22nd International Conference on Computational Linguistics pp. 865-872
- Joel Tetreault, Jennifer Foster and Martin Chodorow. 2010. *Using parse features for preposition selection and error detection*. In Proceedings of the ACL 2010 Conference Short Papers, pp. 353-358
- John Lee and Stephanie Seneff. 2008. *Correcting Misuse of Verb Forms*. In Proceedings 46th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, pp. 175-182
- Junhui Li, Guodong Zhou, Hai Zhao, Qiaoming Zhu, and Peide Qian. 2009. *Improving Nominal SRL in Chinese Language with Verbal SRL Information and*

- Automatic Predicate Recognition*. In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, pp. 1280-1288
- Keisuke Sakaguchi, Yuta Hayashibe, Shuhei Kondo, Lis Kanashiro, Tomoya Mizumoto, Mamoru Komachi and Yuji Matsumoto. 2012. *NAIST at the HOO 2012 Shared Task*. In Proceedings of the Seventh Workshop on Building Educational Applications Using NLP, pp. 281C288.
- Kristina Toutanova, Dan Klein, Christopher Manning and Yoram Singer. 2003. *Feature-Rich Part-of-Speech Tagging with a Cyclic Dependency*. In Proceedings of the 2003 Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics, pp. 252-259
- Li Quan, Aleksandr Kolomyets and Marie-Francine Moens. 2012. *KU Leuven at HOO-2012: a hybrid approach to detection and correction of determiner and preposition errors in non-native English text*. In Proceedings of the Seventh Workshop on Building Educational Applications Using NLP, pp. 263C271.
- Mark Kantrowitz. 2003. *Method and apparatus for analyzing affect and emotion in text*. Patent No. 6,622,140.
- Michael Gamon, Claudia Leacock, Chris Brockett, William B. Dolan, Jianfeng Gao, Dmitriy Belenko and Alexandre Klementiev. 2009. *Using statistical techniques and web search to correct ESL errors*. CALICO Journal, Special Issue on Automatic Analysis of Learner Language, 26(3):491C511.
- Michael Gamon, Jianfeng Gao, Chris Brockett, Alexandre Klementiev, William B. Dolan, Dmitriy Belenko and Lucy Vanderwende. 2008. *Using contextual speller techniques and language modeling for ESL error correction*. In Proceedings of third International Joint Conference on Natural Language Processing Proceedings of the Conference
- Michael Gamon. 2010. *Using mostly native data to correct errors in learners writing: a meta-classifier approach*. In Proceedings of the 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics, pp. 163C171
- Na-Rae Han, Joel Tetreault, Soo-Hwa Lee and Jin-Young Ha. 2010. *Using an error-annotated learner corpus to develop an ESL/EFL error correction system*. In Proceedings of LREC, pp. 763C770
- Na-Rae Han, Martin Chodorow and Claudia Leacock. 2012. *Detecting errors in English article usage by non-native speakers*. Journal of Natural Language Engineering, pp. 115-129
- Rachele De Felice and Stephen G. Puluman. 2008. *A Classifier-Based Approach to Preposition and Determiner Error Correction in L2 English*. In Proceedings of the 22nd International Conference on Computational Linguistics (COLING2008), pp. 169-176
- Randolph Quirk, Sidney Greenbaum, Geoffrey Leech and Jan Svartvik. 1985. *A Comprehensive Grammar of the English Language*. Longman, New York
- Rui Wang, Hai Zhao, Baoliang Lu, Masao Utiyama, and Eiichiro Sumita. 2015 *Bilingual Continuous-Space Language Model Growing for Statistical Machine Translation*. IEEE/ACM Transactions on Audio, Speech, and Language Processing, Vol.23(7): 1209-1220
- Rui Wang, Hai Zhao, Baoliang Lu, Masao Utiyama, and Eiichiro Sumita. 2014. *Neural Network Based Bilingual Language Model Growing for Statistical Machine Translation*. In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, pp. 189-195
- Rui Wang, Masao Utiyama, Isao Goto, Eiichiro Sumita, Hai Zhao, and Baoliang Lu. 2013. *Converting Continuous-Space Language Models into N-gram Language Models for Statistical Machine Translation*. In Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, pp. 845-850
- Ryo Nagata, Edward Whittaker and Vera Sheinman. 2011. *Creating a manually error-tagged and shallow-parsed learner corpus*. In Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, pp. 1210-1219
- Toshikazu Tajiri, Mamoru Komachi and Yuji Matsumoto. 2012. *Tense and aspect error correction for esl learners using global context*. In Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics, pp. 198-202
- Xiaolin Wang, Hai Zhao, and Baoliang Lu. 2013 *Labeled Alignment for Recognizing Textual Entailment*. International Joint Conference on Natural Language Processing, pp. 605-613
- Xuezhe Ma and Hai Zhao. 2012. *Fourth-Order Dependency Parsing*. In Proceedings of the 24th International Conference on Computational Linguistics, pp. 8-15
- Zhongye Jia, Peilu Wang, and Hai Zhao. 2013. *Grammatical Error Correction as Multiclass Classification with Single Model*. In Proceedings of the Seventeenth Conference on Computational Natural Language Learning, pp.74-81
- Zhongye Jia and Hai Zhao. 2014. *A Joint Graph Model for Pinyin-to-Chinese Conversion with Typo Correction*. In Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, pp. 1512-1523

# A Machine Learning Method to Distinguish Machine Translation from Human Translation

Yitong Li<sup>1</sup>, Rui Wang<sup>1,2</sup>, Hai Zhai<sup>1,2</sup> \* †

<sup>1</sup>Center for Brain-Like Computing and Machine Intelligence,  
Department of Computer Science and Engineering,  
Shanghai Jiao Tong University, Shanghai, 200240, China

<sup>2</sup>Key Laboratory of Shanghai Education Commission for Intelligent Interaction  
and Cognitive Engineering, Shanghai Jiao Tong University, Shanghai, 200240, China  
lrnk@sjtu.edu.cn, wangrui.nlp@gmail.com, zhaohai@cs.sjtu.edu.cn

## Abstract

This paper introduces a machine learning approach to distinguish machine translation texts from human texts in the sentence level automatically. In stead of traditional methods, we extract some linguistic features only from the target language side to train the prediction model and these features are independent of the source language. Our prediction model presents an indicator to measure how much a sentence generated by a machine translation system looks like a real human translation. Furthermore, the indicator can directly and effectively enhance statistical machine translation systems, which can be proved as BLEU score improvements.

## 1 Introduction

The translation performance of Statistical Machine Translation (SMT) systems has been improved significantly within this decade. However, it is still incomparable to the human translation (Feng et al., 2012; Li et al., 2012). Most translation text generated by SMT systems can be understood in some

degree but still not good enough. However, a significant proportion of text that exists serious mistakes and even does not make sense, and these text can be easily recognized by human.

It is not difficult to understand the reason why SMT systems generate ill-formed or non-sense sentences. SMT systems combine probability models in a log-linear framework (Och and Ney, 2003), where the systems always attempt to find a sentence with the highest probability from the candidates. However, Language Model (LM), such as  $n$ -gram LM, and reordering model only have limited capacity to represent context, where sentences with local optimum could often be output. Meanwhile, it can be a very different thing for the entire translation sentence due to complicated semantic and pragmatic issues.

Therefore, to improve SMT performance, if poorly translated sentences can be distinguished automatically, it is possible for us to refine these sentences by some extra efforts. In this paper, to order to define the quality of the sentence generated by SMT systems, we borrow the idea from the evaluation of machine translation task, that the more like human translation text, the better the machine translation output is. Considering that the poorly translated sentences show great difference from human text, we compare text generated by SMT systems with human translations. This comparison motivates us to design a predictor to tell whether a sentence is machine generated or human generated. Above all, such a predictor can be treated as a binary classification problem.

In this paper, we use Support Vector Machines (SVMs) (Hearst et al., 1998) to solve such a problem. The benefits of SVMs for text categorization have been identified since it learns well with many

\*Correspondence author.

†Thank all the reviewers for valuable comments and suggestions on our paper. This work was partially supported by the National Natural Science Foundation of China (No. 61170114, and No. 61272248), the National Basic Research Program of China (No. 2013CB329401), the Science and Technology Commission of Shanghai Municipality (No. 13511500200), the European Union Seventh Framework Program (No. 247619), the Cai Yuanpei Program (CSC fund 201304490199 and 201304490171), and the art and science interdiscipline funds of Shanghai Jiao Tong University, No. 14X190040031, and the Key Project of National Society Science Foundation of China, No. 15-ZDA041.

relevant features (Joachims, 1998). In order to find those poorly SMT-translated sentences, we train an SVM-classifier on a feature space. Most features are linguistically motivated only from the target language side. As only target language is concerned, our model will be facilitated of some direct applications.

Among all features, a major part is related to the syntactic parser. The parsing structure of the output sentence is very sensible to the quality of SMT outputs. We therefore especially select these features related to the branching properties of the parse tree. One of the reason is that it had become apparent from failure analysis in (Corston-Oliver et al., 2001) that SMT system output tended to favor right-branching structures over noun compounding.

The remainder of this paper is organized as follows: In Section 2, we will give a quick review on SMT and relevant classification tasks. The SVM approaches and all the features used in our method will be presented in Section 3. Section 4 will give a description on the experiments and an analysis of corresponding results. Last, we will conclude our work in Section 5.

## 2 Related Work

In the classification task part, as our goal is to distinguish sentences with different quality, we are actually working on confidence estimation or automatic evaluation of SMT systems (Doddington, 2002; Papieni et al., 2002; Zhang et al., 2014).

Early work on automatic evaluation of machine translation text estimates the quality at the word level (Gandraber and Foster, 2003; Ueffing and Ney, 2005). Namely,  $n$ -gram features played an important role in translation quality differentiation. However, this paper considers deep level of linguistic features such as those derived from parsing tree instead of  $n$ -gram features.

Liu and Gildea (2005) also used features related to the syntactic parser. Compared with our work, they cared more about detailed syntax properties of the sentences on the parse trees. In this paper, we use less properties but more syntactic structure features.

Corston-Oliver et al. (2001) adopted parse tree related features to evaluating MT. Their work shows a high accuracy in the classification task. However,

the generation of their training and test data should limit to the same SMT system. In this paper, we devote to developing a model that is capable of distinguishing texts generated by multiple sourced SMT systems from human texts. To achieve such an aim, we will introduce quite different types of features such as emotion agreement inside a sentence.

In the statistical machine translation systems part, the performance is depended on the LM and translation model. Traditional Back-off  $n$ -gram LMs (BNLMs) (Chen and Goodman, 1996; Chen and Goodman, 1999; Stolcke, 2002) have been widely used for probability estimation and BNLMs also show up in many other NLP tasks (Jia and Zhao, 2014; Zhang et al., 2012; Xu and Zhao, 2012). Recently, a better probability estimation method, Continuous-Space Language Models (CSLMs), especially Neural Network Language Models (NNLMs) (Bengio et al., 2003; Schwenk et al., 2006; Schwenk, 2007; Le et al., 2011) are being used in SMT tasks (Son et al., 2010; Son et al., 2012; Wang et al., 2013; Wang et al., 2015; Wang et al., 2014). Also, Neural Network Translation Models (NNTMs) show a success in SMT (Kalchbrenner and Blunsom, 2013; Blunsom et al., 2014; Devlin et al., 2014). However, the high cost of CSLMs makes it difficult to decoding directly. This leads to a  $n$ -best reranking method which is available for our paper (Schwenk et al., 2006; Son et al., 2012).

## 3 The Proposed Approach

In this Section, we present a machine learning method to distinguish poor translated sentences from good ones.

### 3.1 Support Vector Machine

For text classification tasks, Many approaches have been proposed (Sebastiani, 2002). Among these approaches, SVM has shown widely applications (Joachims, 1998; Joachims, 1999; Joachims, 2002; Tong and Koller, 2002). And in following subsection we will introduces how to formalize the proposed task.

The training corpus for the classifier includes  $l$  human translation sentences as positive samples and  $l$  corresponding SMT outputs as negative samples. For a sentence  $S$ , it can be represented by an

$N$ -dimensional feature vector  $V \{v_1, v_2, \dots, v_N\}$ , where  $N$  is total number of all the features, and in most cases,  $v_i$  is a real number feature normalized by the length  $L_S$  of sentence  $S$ .

With the above training corpus, we will train an SVM classifier with linear kernel. The SVM prediction function is defined as the following:

$$predict(S) = \begin{cases} +1, & h(S) \geq 0 \\ -1, & h(S) < 0 \end{cases}$$

where

$$h(S) = w_1v_1 + w_2v_2 + \dots + w_Nv_N$$

In this paper, Liblinear (Fan et al., 2008) is adopted as our SVM implementation and the parameter soft margin width is optimized over a small development set.

### 3.2 Features

In this subsection, we will present our feature collections.

Considering that only the properties of target language are involved in our expectation, we decide to use specific types of linguistic features to present the quality of the sentence.

A very important type of linguistic features is directly linked to syntactic structure of sentence. When getting the parse tree of a sentence, we can exploit a number of available properties, such as sentence structure and the densities of constituent types, to design as our features.

For parser implementation, we use Stanford Lexicalized Parser version 3.3.1. (De Marneffe et al., 2006). Figure 1 gives an example of a parse tree.

The features related to the parse tree are as the following<sup>1</sup>:

- number of right-branching nodes for all constituent types and for Noun Phrases (NPs).

Using Figure 1 as an example, there are 13 right-branching nodes for all constituent types in colorful frames, including one NP in the red frame. Normalized by the length of the sentence 16, feature scores are respectively 0.8125 and 0.0625.

<sup>1</sup>In default, all the following counting numbers for feature score computation are normalized by the length of the sentence.

- number of left-branching nodes for all constituent types and for NPs
- number of pre-modifiers, adjectives before nouns, for all constituent types and for NPs
- number of post-modifiers, adjectives after nouns, for all constituent types and for NPs
- branching index, the number of right-branching nodes minus number of left-branching nodes, for all constituent types and for NPs
- branching weight index, number of tokens covered by right-branching nodes minus number of tokens covered by left-branching nodes, for all constituent types and for NPs
- modification index, the number of pre-modifiers minus the number of post-modifiers, for all constituent types and for NPs
- modification weight index, length in tokens of all pre-modifiers minus length in tokens of all post-modifiers, for all constituent types and for NPs

We also consider density of function words as well as the pronouns, where SMT systems make mistakes frequently. All densities are computed by counting the words with sentence length normalization:

- overall function word density
- density of determiners
- density of quantifiers
- density of pronouns
- density of prepositions
- density of punctuation marks
- density of auxiliary verbs
- density of conjunctions
- density of different pronoun: Wh-, 1st, 2nd, and 3rd person pronouns

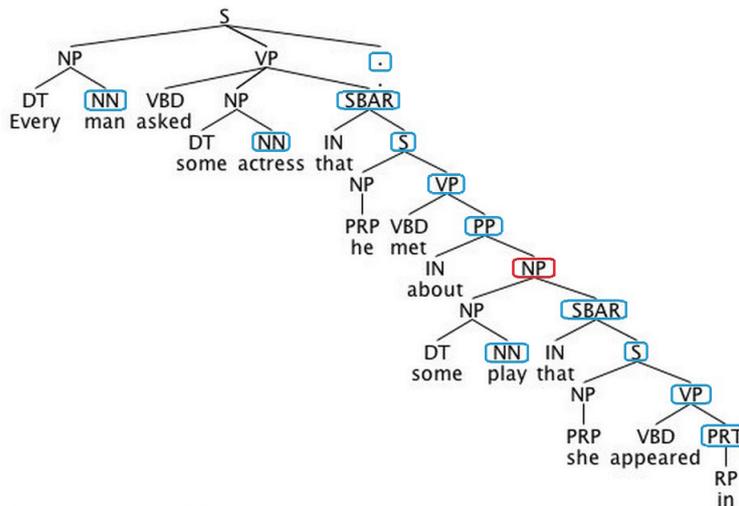


Figure 1: An Example of Parse Tree

The presence of out of vocabulary (OOV) word usually make situations more complicated. Also, problem like subject-verb disagreement are easy to be detected. Therefore, we give a group of lexical-level features:

- number of OOV words
- types of the immediate children of the root
- subject-verb disagreement

In additional, we score emotion agreement inside a sentence as features. This is motivated by the observation that a reasonable sentence should have a consistent emotion strength among different words. To evaluate such agreement, we build a dictionary  $D_{emotion}$  especially for emotion words in advance, in which each word  $s_i$  can be scored from  $-3$  to  $+3$ . We score all the words into these categories with a linear model to describe the strength of emotion. To a sentence, the average scoring and standard deviation will be considered:

- $\mu_{emotion}(S)$
- $\sigma_{emotion}(S)$

where  $S$  is a sentence with length  $len$ .

Finally, sizes of the following constituents are measured:

- sentence length

- parse tree depth
- maximal and average NP length
- maximal and average Adjective Phrase (ADJP) length
- maximal and average Prepositional Phrase (PP) length
- maximal and average Adverb Phrase (ADVP) length

## 4 Experiment

### 4.1 Classification

In this subsection, we will give experiment details of the prediction model.

In all of our experiments, the default settings<sup>2</sup> of Moses (Koehn et al., 2007) and GIZA++ (Och and Ney, 2003) are used for system building. For each SMT system, a 5-gram LM (Chen and Goodman, 1996) is trained on the target side of training set using IRST LM Toolkit.

We use four language pairs from version 7 of the Europarl corpus<sup>3</sup> (Koehn, 2005) as our experiment data and train four SMT systems, respec-

<sup>2</sup>In this paper, we build only phrase-based SMT for experiment implementation. However, we believe this method is feasible for other SMT systems, such as syntax-based SMT.

<sup>3</sup><http://www.statmt.org/europarl/>

tively: French-English, German-English, Italian-English and Danish-English.

Considering the consistency of system and convenience of analysis, all these four systems use English as target language. We use these four systems to generate translation text.

We randomly pick 5K sentences from the French corpus, noted as  $F1(5K)$ , and translated into English sentences  $E1(5K)$  as our negative samples, by SMT system. The corresponding English part  $E1'$  of  $F1$  is used as the positive samples.  $\{E1, E1'\}$  forms the required training set. Then, we randomly pick 10K sentences from each of French  $F2(10K)$ , German  $G2(10K)$ , Italian  $I2(10K)$  and Danish  $D2(10K)$  corpora and translate them into English text  $E2(40K)$ . Another 40K sentences are extracted from English  $E2'(40K)$ .  $\{E2, E2'\}$  forms a multi-model-translated-text test set.  $F2$  has no cover with  $F1$ .

The prediction results are shown in Table 1:

Data Set	Accuracy
Training set	92.3%
Test set	74.2%

Table 1: Classification Accuracy

### 4.2 Feedback to SMT system

One direct application of our prediction model is to provide feedback to SMT systems.

We select the French-English SMT system that we built above as our baseline. For the sake of modifying the system as little as possible, we consider an  $n$ -best list and reranking method on the output candidates of the baseline.

We make a slight change on the prediction model so that it can give a confidence score between 0 and 1 on each sentence. The nearer with 1 its score is, the better the sentence will be. For each SMT output sentence, we choose a 1000-candidate<sup>4</sup> list sorted by the baseline, and score them by our prediction model. We check each candidate by the original sort, and find out the first candidate whose score is greater than a threshold  $H$  as our new output.<sup>5</sup> In case that

<sup>4</sup>This is an empirical value.

<sup>5</sup>We considered directly adding SVM score as a new feature into SMT system, however our current method shown in this paper gets better results. Also, this method is more efficient.

no candidates satisfy the condition, we simply give the origin output.

In our experiment, we set  $H$  empirically. Table 2 shows the 1.6 BLEU score refined by our method.

MT System	BLEU Score
Baseline	23.5
Refined $H = 0.6$	24.7
Refined $H = 0.7$	<b>25.1</b>
Refined $H = 0.8$	23.9

Table 2: BLEU scores

### 4.3 Discussion

We will discuss how our method works by examples. Table 3 shows a translation and refined example.

<i>S</i>	Quelle que soit la bonne réponse, la question est que la détermination des mesures à prendre concernant la race représente un problème dominant dans la politique américaine.
<i>T</i>	Whatever the answer, the question is the determination of the action on the race is a dominant issue in American politics.
<i>R</i>	Whatever the answer, the question is that determining what to do about race is a dominant issue in American politics.
<i>Ref</i>	Regardless of the correct answer, the point is that determining what to do about race is a dominant issue in US politics.

Table 3: A Translation Example. *S*: Source, *T*: Target, *R*: Refined, and *Ref*: Reference

According to the analysis, the parse tree structure of output  $T$  is seriously right-deviated, while sentence  $R$  has a more balance tree structure. Our prediction model will consider  $R$  as a good translation but  $T$  as a bad one. When reordering candidates, our algorithm successfully selects  $R$  as output instead of  $T$ . In addition, compared with reference sentence, we see that  $R$  is an even better translation.

## 5 Conclusion

In this paper, we present an indicator that using linguistic features to train an SVM classifier to distinguish poor SMT sentences from good ones. We use

single-MT-model-generated text as training data and multi-MT-model-generated text as test data to show the stability of our method. With the help of a series of features derived from parse tree, emotion agreement and lexical features, our classifier gives acceptable accuracy. In addition, we show that such a predicator can effectively enhance the corresponding SMT task.

## References

- Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Janvin. 2003. A neural probabilistic language model. *The Journal of Machine Learning Research*, 3:1137–1155.
- Phil Blunsom, Edward Grefenstette, Nal Kalchbrenner, et al. 2014. A convolutional neural network for modelling sentences. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*. Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics.
- Stanley F Chen and Joshua Goodman. 1996. An empirical study of smoothing techniques for language modeling. In *Proceedings of the 34th annual meeting on Association for Computational Linguistics*, pages 310–318. Association for Computational Linguistics.
- Stanley F. Chen and Joshua Goodman. 1999. An empirical study of smoothing techniques for language modeling. *Computer Speech & Language*, volume 13(4):359–393(35).
- Simon Corston-Oliver, Michael Gamon, and Chris Brockett. 2001. A machine learning approach to the automatic evaluation of machine translation. In *Proceedings of the 39th Annual Meeting on Association for Computational Linguistics*, pages 148–155. Association for Computational Linguistics.
- Marie-Catherine De Marneffe, Bill MacCartney, Christopher D Manning, et al. 2006. Generating typed dependency parses from phrase structure parses. In *Proceedings of the International Conference on Language Resources and Evaluation*, volume 6, pages 449–454.
- Jacob Devlin, Rabih Zbib, Zhongqiang Huang, Thomas Lamar, Richard Schwartz, and John Makhoul. 2014. Fast and robust neural network joint models for statistical machine translation. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, volume 1, pages 1370–1380.
- George Doddington. 2002. Automatic evaluation of machine translation quality using n-gram co-occurrence statistics. In *Proceedings of the second international conference on Human Language Technology Research*, pages 138–145. Morgan Kaufmann Publishers Inc.
- Rong-En Fan, Kai-Wei Chang, Cho-Jui Hsieh, Xiang-Rui Wang, and Chih-Jen Lin. 2008. Liblinear: A library for large linear classification. *The Journal of Machine Learning Research*, 9:1871–1874.
- Yang Feng, Dongdong Zhang, Mu Li, Ming Zhou, and Qun Liu. 2012. Hierarchical chunk-to-string translation. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers-Volume 1*, pages 950–958. Association for Computational Linguistics.
- Simona Gandrabur and George Foster. 2003. Confidence estimation for translation prediction. In *Proceedings of the 7th conference on Natural language learning at HLT-NAACL 2003-Volume 4*, pages 95–102. Association for Computational Linguistics.
- Marti A. Hearst, Susan T Dumais, Edgar Osman, John Platt, and Bernhard Scholkopf. 1998. Support vector machines. *Intelligent Systems and their Applications, IEEE*, 13(4):18–28.
- Zhongye Jia and Hai Zhao. 2014. A joint graph model for pinyin-to-chinese conversion with typo correction. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, pages 1512–1523.
- Thorsten Joachims. 1998. *Text categorization with support vector machines: Learning with many relevant features*. Springer.
- Thorsten Joachims. 1999. Transductive inference for text classification using support vector machines. In *Proceedings of the 16th International Conference on Machine Learning*, volume 99, pages 200–209.
- Thorsten Joachims. 2002. *Learning to classify text using support vector machines: Methods, theory and algorithms*. Kluwer Academic Publishers.
- Nal Kalchbrenner and Phil Blunsom. 2013. Recurrent continuous translation models. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1700–1709.
- Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, et al. 2007. Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th annual meeting of the Association for Computational Linguistics on interactive poster and demonstration sessions*, pages 177–180. Association for Computational Linguistics.
- Philipp Koehn. 2005. Europarl: A parallel corpus for statistical machine translation. In *Machine Translation summit*, volume 5, pages 79–86. Citeseer.
- Hai-Son Le, Ilya Oparin, Alexandre Allauzen, Jean-Luc Gauvain, and François Yvon. 2011. Structured output layer neural network language model. In *2011*

- IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 5524–5527. IEEE.
- Junhui Li, Zhaopeng Tu, Guodong Zhou, and Josef van Genabith. 2012. Head-driven hierarchical phrase-based translation. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Short Papers-Volume 2*, pages 33–37. Association for Computational Linguistics.
- Ding Liu and Daniel Gildea. 2005. Syntactic features for evaluation of machine translation. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pages 25–32.
- Franz Josef Och and Hermann Ney. 2003. A systematic comparison of various statistical alignment models. *Computational linguistics*, 29(1):19–51.
- Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on association for computational linguistics*, pages 311–318. Association for Computational Linguistics.
- Holger Schwenk, Daniel Dchelotte, and Jean-Luc Gauvain. 2006. Continuous space language models for statistical machine translation. In *Proceedings of the COLING/ACL on Main conference poster sessions*, pages 723–730. Association for Computational Linguistics.
- Holger Schwenk. 2007. Continuous space language models. *Computer Speech & Language*, 21(3):492–518.
- Fabrizio Sebastiani. 2002. Machine learning in automated text categorization. *ACM computing surveys (C-SUR)*, 34(1):1–47.
- Le Hai Son, Alexandre Allauzen, Guillaume Wisniewski, and François Yvon. 2010. Training continuous space language models: Some practical issues. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 778–788. Association for Computational Linguistics.
- Le Hai Son, Alexandre Allauzen, and François Yvon. 2012. Continuous space translation models with neural networks. In *Proceedings of the 2012 conference of the north american chapter of the association for computational linguistics: Human language technologies*, pages 39–48. Association for Computational Linguistics.
- Andreas Stolcke. 2002. Srilm – an extensible language modeling toolkit. In *Proceedings of the 7th International Conference on Spoken Language Processing*, pages 901–904.
- Simon Tong and Daphne Koller. 2002. Support vector machine active learning with applications to text classification. *The Journal of Machine Learning Research*, 2:45–66.
- Nicola Ueffing and Hermann Ney. 2005. Application of word-level confidence measures in interactive statistical machine translation. In *Proceedings of the 10th Annual Conference of the European Association for Machine Translation*, pages 262–270.
- Rui Wang, Masao Utiyama, Isao Goto, Eiichiro Sumita, Hai Zhao, and Bao-Liang Lu. 2013. Converting continuous-space language models into n-gram language models for statistical machine translation. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 845–850, Seattle, Washington, USA, October. Association for Computational Linguistics.
- Rui Wang, Hai Zhao, Bao-Liang Lu, Masao Utiyama, and Eiichiro Sumita. 2014. Neural network based bilingual language model growing for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, pages 189–195, Doha, Qatar, October. Association for Computational Linguistics.
- Rui Wang, Hai Zhao, Bao-Liang Lu, Masao Utiyama, and Eiichiro Sumita. 2015. Bilingual continuous-space language model growing for statistical machine translation. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*.
- Qiongkai Xu and Hai Zhao. 2012. Using deep linguistic features for finding deceptive opinion spam. In *Proceedings of the 24th International Conference on Computational Linguistics*, pages 1341–1350. Cite-seer.
- Xiaotian Zhang, Hai Zhao, and Cong Hui. 2012. A machine learning approach to convert ccgbank to penn treebank. In *Proceedings of the 24th International Conference on Computational Linguistics*, pages 535–542.
- Jingyi Zhang, Masao Utiyama, Eiichiro Sumita, and Hai Zhao. 2014. Learning hierarchical translation spans. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, pages 183–188.