Co-occurrence Graph Based Iterative Bilingual Lexicon Extraction From Comparable Corpora

Diptesh Chatterjee and **Sudeshna Sarkar** and **Arpit Mishra**
Department of Computer Science and Engineering
Indian Institute of Technology Kharagpur
{diptesh,sudeshna,arpit}@cse.iitkgp.ernet.in

Abstract

This paper presents an iterative algorithm for bilingual lexicon extraction from comparable corpora. It is based on a bag-of-words model generated at the level of sentences. We present our results of experimentation on corpora of multiple degrees of comparability derived from the FIRE 2010 dataset. Evaluation results on 100 nouns shows that this method outperforms the standard context-vector based approaches.

1 Introduction

Bilingual dictionaries play a pivotal role in a number of Natural Language Processing tasks like Machine Translation and Cross Lingual Information Retrieval(CLIR). Machine Translation systems often use bilingual dictionaries in order to augment word and phrase alignment (Och and Ney, 2003). CLIR systems use bilingual dictionaries in the query translation step (Grefenstette, 1998). However, high coverage electronic bilingual dictionaries are not available for all language pairs. So a major research area in Machine Translation and CLIR is bilingual dictionary extraction. The most common approach for extracting bilingual dictionary is applying some statistical alignment algorithm on a parallel corpus. However, parallel corpora are not readily available for most language pairs. Also, it takes a lot of effort to actually get the accurate translations of sentences. Hence, constructing parallel corpora involves a lot of effort and time. So in recent years, extracting bilingual dictionaries from comparable corpora has become an important area of research. Comparable corpora consist of documents on similar topics in different languages. Unlike parallel corpora, they are not sentence aligned. In fact, the sentences in one language do not have to be the exact translations of the sentence in the other language. However, the two corpora must be on the same domain or topic. Comparable corpora can be obtained more easily than parallel corpora. For example, a collection of news articles from the same time period but in different languages can form a comparable corpora. But after careful study of news articles in English and Hindi published on same days at the same city, we have observed that along with articles on similar topics, the corpora also contain a lot of articles which have no topical similarity. Thus, the corpora are quite noisy, which makes it unsuitable for lexicon extraction. Thus another important factor in comparable corpora construction is the degree of similarity of the corpora.

Approaches for lexicon extraction from comparable corpora have been proposed that use the bag-of-words model to find words that occur in similar lexical contexts (Rapp, 1995). There have been approaches proposed which improve upon this model by using some linguistic information (Yuu and Tsujii, 2009). However, these require some linguistic tool like dependency parsers which are not commonly obtainable for resource-poor languages. For example, in case of Indian languages like Hindi and Bengali, we still do not have good enough dependency parsers. In this paper, we propose a word co-occurrence based approach for lexicon extraction from comparable corpora using English and Hindi as the source and target languages respectively. We do not use any language-

Proceedings of the 4th International Workshop on Cross Lingual Information Access at COLING 2010, pages 35–42,
Beijing, August 2010

specific resource in our approach.

We did experiments with 100 words in English,and show that our approach performs significantly better than the the Context Heterogeneity approach (Fung, 1995). We show the results over corpora with varying degrees of comparability.

The outline of the paper is as follows. In section 2, we analyze the different approaches for lexicon extraction from comparable corpora. In section 3, we present our algorithm and the experimental results. In section 4, we present an analysis of the results followed by the conclusion and future research directions in section 5.

2 Previous Work

One of the first works in the area of comparable corpora mining was based on word co-occurrence based approach (Rapp, 1995). The basic assumption behind this approach was two words are likely to occur together in the same context if their joint probability of occurrence in a corpus exceeds the probability that the words occur randomly. In his paper, Rapp made use of a similarity matrix and using a joint probability estimate determined the word maps. However this approach did not yield significantly good results.

The "Context Heterogeneity" approach was one of the pioneering works in this area. It uses a 2-dimensional context vector for each word based on the right and left context. The context vector depended on how many distinct words occur in the particular context and also the unigram frequency of the word to be translated. Euclidean distance between context vectors was used as a similarity measure.

Another approach used Distributed Clustering of Translational Equivalents for word sense acquisition from bilingual comparable corpora (Kaji, 2003). However, the major drawback of this paper is the assumption that translation equivalents usually represent only one sense of the target word. This may not be the case for languages having similar origin, for example, Hindi and Bengali.

Approaches using context information for extracting lexical translations from comparable corpora have also been proposed (Fung and Yee, 1998; Rapp, 1999). But they resulted in very poor coverage. These approaches were improved upon by extracting phrasal alignments from comparable corpora using joint probability SMT model (Kumano et al., 2007) .

Another proposed method uses dependency parsing and Dependency Heterogeneity for extracting bilingual lexicon (Yuu and Tsujii, 2009) . This approach was similar to that of Fung, except they used a dependency parser to get the tags for each word and depending on the frequency of each tag they defined a vector to represent each word in question. Here too, Euclidean similarity was used to compute the similarity between two words using their context vectors. However, this method is dependent on availability of a dependency parser for the languages and is not feasible for languages for which resources are scarce.

3 Bilingual Dictionary Extraction Using Co-occurrence Information

3.1 Motivation

The Context Heterogeneity and Dependency Heterogeneity approaches suffer from one major drawback. They do not use any kind of information about how individual words combine in a particular context to form a meaningful sentence. They only use some statistics about the number of words that co-occur in a particular context or the number of times a word receives a particular tag in dependency parsing. So, we wished to study if the quality of dictionary extracted would improve if we consider how individual words co-occur in text and store that information in the form of a vector, with one dimension representing one word in the corpus. One important point to note here is that the function words in a language are usually very small in number. If we need to construct a dictionary of function words in two languages, that can be done without much effort manually. Also, the function words do not play an important role in CLIR applications, as they are usually stripped off.

Our algorithm is based on the intuition that words having similar semantic connotations occur together. For example, the words "bread" is more likely to occur with "eat" than with "play". Our algorithm uses this distribution of co-occurrence frequency along with a small initial seed dictio-

nary to extract words that are translations of one another. We define a co-occurrence vector of words in both the languages, and also record the number of times two words co-occur. To find the translation for word W_x, we check for the words co-occurring with W_x such that this word already has a map in the other language, and compute a scoring function using all such words co-occurring with W_x. In short, we use the already existing information to find new translations and add them to the existing lexicon to grow it. Below is a snapshot of a part of the data from one of our experiments using the FIRE 2010[1] corpus. For each word in English and Hindi, the co-occurrence data is expressed as a list of tuples. Each tuple has the form **(word, co-occurrence frequency)**. For the Hindi words, the English meaning has been provided in parenthesis. For the seed lexicon and final lexicon, the format is **(source word, target word, strength)**.
English:

1. **teacher**:{(training,49),(colleges,138), (man,22)}
2. **car**:{(drive,238),(place,21)}
3. **drive**:{(car,238),(steer,125),(city,12), (road,123)}

Hindi:

1. **ghar(home)**:{(khidki(window),133),(makAn (house),172), (rAstA(road),6)}
2. **gAdi(car)**:{(rAsta,92),(chAlak(driver),121), (signal,17)}
3. **shikshaka(teacher)**:{(vidyalaya(school),312), (makAn(house),6)}

Seed lexicon:

1. (colleges,vidyalaya,0.4)
2. (colleges,mahavidyalaya(college),0.6)
3. (car,gAdi,1.0)

The following is a snapshot from the final results given by the algorithm:

1. (car,gAdi,1.0)
2. (teacher,shikshak,0.62)
3. (teacher, vidyalaya,0.19)
4. (road, rAsta, 0.55)

3.2 The Algorithm

For extracting bilingual lexicon, we have not considered the function words of the two languages. In order to filter out the function words, we have made use of the assumption that content words usually have low frequency in the corpus, whereas function words have very high frequency. First, we define some quantities:

Let the languages be **E** and **H**.
W_e = Set of words in **E** $= \{e_1, e_2, ..., e_N\}$
W_h = Set of words in **H** $= \{h_1, h_2, ..., h_M\}$

$|W_e| = N$
$|W_h| = M$

MAP = Initial map given
$= \{(e_i, h_j, w_{ij}) | w_{ij} = wt(e_i, h_j), e_i \in W_e, h_j \in W_h\}$

E_M = Set of words in **E** which are included in entries of MAP

H_M = Set of words in **H** which are included in entries of MAP

$Co\_occ(x)$ = Set of words which co-occur with word x

$$Co\_occ'(x) = \begin{cases} Co\_occ(x) \cap E_M & \text{if } x \in W_e \\ Co\_occ(x) \cap H_M & \text{if } x \in W_h \end{cases}$$

$Wt_e(x) = \{W_{ey} | y \in W_e \text{ and } y \in Co\_occ(x)\}$
$Wt_h(x) = \{W_{hy} | y \in W_h \text{ and } y \in Co\_occ(x)\}$

Given a comparable corpus, we follow the following steps of processing:

1. A sentence segmentation code is run to segment the corpus into sentences.
2. The sentence-segmented corpus is cleaned of all punctuation marks and special symbols by replacing them with spaces.

[1] Forum For Information Retrieval
http://www.isical.ac.in/∼clia/index.html

Algorithm 1 Algorithm to Extract Bilingual Dictionary by using word Co-occurrence Information

```
repeat
  for $e_i \in W_e$ do
    for $h_j \in W_h$ do
      if $(e_i, h_j, 0) \in MAP$ then
        $wt(e_i, h_j) = \frac{\sum_{e \in Co\_occ'(e_i)} \sum_{h \in Co\_occ'(h_j)} (W_{ij} W_{ee_i} W_{hh_j})}{\sum_{e \in Co\_occ'(e_i)} \sum_{h \in Co\_occ'(h_j)} (W_{ee_i} W_{hh_j})}$
      end if
    end for
  end for
  Select the pair with highest value of $wt(e_i, b_j)$ and add it to the existing map and normalize
until termination
```

3. The collection frequency of all the terms are computed and based on a threshold, the function words are filtered out.

4. The co-occurrence information is computed at sentence-level for the remaining terms. In a sentence, if words w_i and w_j both occur, then $w_i \in Co\_occ(w_j)$ and vice versa.

5. Since we can visualize the co-occurrence information in the form of a graph, we next cluster the graph into C clusters.

6. From each cluster C_i, we choose some fixed number number of words and manually find out their translation in the target language. This constitutes the initial map.

7. Next we apply Algorithm 1 to compute the word maps.

The time complexity of the algorithm is $O(IM^2N^2)$, where I is the number of iterations of the algorithm.

3.3 Corpus Construction

The corpora used for evaluating our algorithm were derived from the FIRE 2010 English and Hindi corpora for the ad-hoc retrieval task. These corpora contained news articles spanning over a time period of three years from two Indian newspapers, "The Dainik Jagaran" in Hindi and "The Telegraph" in English. However, due to the extreme level of variation of the topics in these corpora, we applied a filtering algorithm to select a subset of the corpora.

Our approach to make the text similar involved reducing the corora based on matching Named Entities. Named Entities of English and Hindi corpus were listed using LingPipe[2] and a Hindi NER system built at IIT Kharagpur(Saha et al., 1999). The listed Named Entities of the two corpora were compared to find the matching Named Entities. Named Entities in Hindi Unicode were converted to iTRANS[3] format and matched with English Named Entities using edit distance. Unit cost was defined for each insert and delete operation. Similar sounding characters like 's', 'c','a', 'e' etc were assigned a replacement cost of 1 and other characters were assigned a replacement cost of 2. Two Named Entities were adjudged matching if:
$(2 * Cost)/(WL_h + WL_e) < 0.5$
where,
WL_h = Length of Hindi word
WL_e = Length of English word
Using this matching scheme, accuracy of matching of Hindi and English Named Entities was found to be $> 95\%$. It was observed that there are large number of Named Entities with small frequency and few Named Entities with large frequency. So a matching list was prepared which contained only those Named Entities which had frequency larger than a $\sqrt{MaxFreq}$. This ensured that matching list had words with high frequency in both corpus.So English words with frequency larger than 368 and Hindi words with frequency larger than 223 were considered for matching. Based on this matching list, the two

[2] http://alias-i.com/lingpipe/
[3] http://www.aczoom.com/itrans/

Language	Total NE	Unique NE	NE with freq larger than $\sqrt{MaxFreq}$	NE Matched	Total No of docs	% of NE covered	
						According to Zipf's Law	In the actual corpus
Hindi	1195474	37606	686	360	54271	63.0%	74.3%
English	5723292	137252	2258	360	87387	65.2%	71.0%

Table 1: Statistics of the main corpora used for extraction

Corpus	Max Freq Word	Max Freq	$\sqrt{MaxFreq}$
Hindi	bharat	50072	223
English	calcutta	135780	368

Table 2: Criteria used for thresholding in the two corpora

Matching % of NE per document	Total documents in corpora	
	Hindi	English
$> 10\%$	34694	16950
$> 20\%$	14872	4927
$> 30\%$	2938	1650

Table 3: Statistics of extracted corpora

corpora were reduced by including only those files each of which contained more than a certain fixed percentage of total matching Named Entities. The corpus statistics are provided in tables 1, 2 and 3.
We assume that distribution of Named Entities follows Zipf's law (Zipf, 1949). And analysis shows that Named Entities with frequency greater than the chosen threshold lead to high coverage both theoretically and in practice (Table 1). Hence, the threshold was chosen as $\sqrt{MaxFreq}$. The differences in the theoretical and actual values can be attributed to the poor performance of the NER systems, especially the Hindi NER system, whose output contained a number of false positives.

3.4 Experimental Setup

The languages we used for our experiments were English and Hindi. English was the source language and Hindi was chosen as the target. For our experiments, we used a collection frequency threshold of 400 to filter out the function words. The words having a collection frequency more than 400 were discarded. This threshold was obtained manually by "Trial and Error" method in order to perform an effective function word filtering. For each corpora, we extracted the co-occurrence information and then clustered the co-occurrence graph into 20 clusters. From each cluster we chose 15 words, thus giving us an overall initial seed dictionary size of 300. We ran the algorithm for 3000 iterations.
For graph clustering, we used the Graclus system (Dhillon et al., 2007) which uses a weighted kernel k-means clustering algorithm at various levels of coarseness of the input graph.

3.5 Evaluation Method and Results

For evaluation, we have used the Accuracy and MMR measure (Voorhees, 1999). The measures are defined as follows:

$$Accuracy = \frac{1}{N}\sum_{i=1}^{N} t_i$$

$$\text{where, } t_i = \begin{cases} 1 & \text{if correct translation in top } n \\ 0 & \text{otherwise} \end{cases}$$

$$MMR = \frac{1}{N}\sum_{i=1}^{N} \frac{1}{rank_i}$$

$$\text{where, } rank_i = \begin{cases} r_i & \text{if } r_i \leq n \\ 0 & \text{otherwise} \end{cases}$$

n means top n evaluation

r_i means rank of correct translation in top n ranking

N means total number of words used for evaluation

For our experiments, we have used:

Corpus	Context Heterogeneity		Co-occurrence	
	Acc	MMR	Acc	MMR
> 10%	0.14	0.112	0.16	0.135
> 20%	0.21	0.205	0.27	0.265
> 30%	0.31	0.285	0.35	0.333

Table 4: Comparison of performance between Context Heterogeneity and Co-occurrence Approach for manual evaluation

$n = 5$
$N = 100$

The 100 words used for evaluation were chosen randomly from the source language.
Two evaluation methods were followed - manual and automated. In the manual evaluation, a person who knows both English and Hindi was asked to find the candidate translation in the target language for the words in the source language. Using this gold standard map, the Accuracy and MMR values were computed.
In the second phase (automated), lexicon extracted is evaluated against English to Hindi wordnet[4]. The evaluation process proceeds as follows:

1. Hashmap is created with English words as keys and Hindi meanings as values.

2. English words in the extracted lexicon are crudely stemmed so that inflected words match the root words in the dictionary. Stemming is done by removing the last 4 characters, one at a time and checking if word found in dictionary.

3. Accuracy and MMR are computed.

As a reference measure, we have used Fung's method of Context Heterogeneity with a context window size of 4. The results are tabulated in Tables 4 and 6. We can see that our proposed algorithm shows a significant improvement over the Context Heterogeneity method. The degree of improvement over the Context Heterogeneity is summarized in Tables 5 and 7. For auto evaluation, We see that the proposed approach shows the maximum improvement (83.33% in Accuracy and 66.67% in MMR) in performance when the corpus size is medium. For very large (too general) corpora, both the approaches give identical result while for very small (too specific) corpora, the proposed approach gives slightly better results than the reference.
The trends are similar for manual evaluation. Once again, the maximum improvement is observed for the medium sized corpus (> 20%). However, in this evaluation system, the proposed approach performs much better than the reference even for the large (more general) corpora.

Corpus	Accuracy	MMR
> 10%	↑ 14.28%	↑ 20.53%
> 20%	↑ 28.57%	↑ 29.27%
> 30%	↑ 12.9%	↑ 16.84%

Table 5: Degree of improvement shown by Co-occurrence approach over Context Heterogeneity for manual evaluation

Corpus	Context Heterogeneity		Co-occurrence	
	Acc	MMR	Acc	MMR
> 10%	0.05	0.08	0.05	0.08
> 20%	0.06	0.06	0.11	0.10
> 30%	0.13	0.11	0.15	0.13

Table 6: Comparison of performance between Context Heterogeneity and Co-occurrence Approach for auto-evaluation

Corpus	Accuracy	MMR
> 10%	0.0%	0.0%
> 20%	↑ 83.33%	↑ 66.67%
> 30%	↑ 15.38%	↑ 18.18%

Table 7: Degree of improvement shown by Co-occurrence approach over Context Heterogeneity for auto-evaluation

[4] Downloadable from http://sanskritdocuments.org/hindi/dict/eng-hin-itrans.html

4 Discussion

The co-occurrence based approach used in this paper is quite a simple approach in the sense that it does not make use of any kind of linguistic information. From the aforementioned results we can see that a model based on simple word co-occurrence highly outperforms the “Context Heterogeneity” model in almost all the cases. One possible reason behind this is the amount of information captured by our model is more than that captured by the “Context Heterogeneity” model. “Context Heterogeneity” does not model actual word-word interactions. Each word is represented by a function of the number of different contexts it can occur in. However, we represent the word by a co-occurrence vector. This captures all possible contexts of the word. Also, we can actually determine which are the words which co-occur with any other word. So our model captures more semantics of the word in question than the “Context Heterogeneity” model, thereby leading to better results. Another possible factor is the nature in which we compute the translation scores. Due to the iterative nature of the algorithm and since we normalize after each iteration, some of the word pairs that received unduly high score in an earlier iteration end up having a substantially low score. However, since the “Context Heterogeneity” does only a single pass over the set of words, it fails to tackle this problem.

The seed dictionary plays an important role in our algorithm. A good seed dictionary gives us some initial information to work with. However, since “Context Heterogeneity” does not use a seed dictionary, it loses out on the amount of information initially available to it. Since the seed dictionary size for our approach is quite small, it can be easily constructed manually. However, how the seed dictionary size varies with corpus size is an issue that remains to be seen.

Another important factor in our algorithm is the way in which we have defined the co-occurrence vectors. This is not the same as the context vector that we define in case of Context Heterogeneity. In a windowed context vector, we fail to capture a lot of dependencies that might be captured using a sentence-level co-occurrence. This problem is especially more visible in case of free-word-order languages like the Indo-European group of languages. For these languages, a windowed context vector is also likely to introduce many spurious dependencies. Since Hindi is a language of this family, our algorithm captures many more correct semantic dependencies than Context Heterogeneity algorithm, resulting in better preformance.

Another strong point of our proposed approach is the closeness of the values of Accuracy and MMR. This shows that the translation candidates extracted by our algorithm are not only correct, but also the best translation candidate gets the highest score with high probability. This is a very important factor in Machine Translation systems, where a more accurate dictionary would give us an improved performance.

A noticeable point about the evaluation scores is the difference in scores given by the automated system and the manual system. This can be attributed to synonymy and spelling errors. In the target language Hindi, synonymy plays a very important part. It is not expected that all synonyms of a particular word may be present in an online dictionary. In such cases, the manual evaluator marks a translation pair as True, whereas the automated system marks it as False. Instances of spelling errors have also been found. For example, for the word ”neighbors”, the top translation provided by the system was ”paDosana”(female neighbor). If we consider root form of words, this is correct. But the actual translation should be ”paDosiyAn”(neighbors, may refer to both male and female). Thus the auto evaluation system tags it as False, whereas the manual evaluator tags it as True. There are many more such occurrences throughout.

Apart from that, the manual evaluation process has been quite relaxed. Even if the properties like tense, number of words does not match, as long as the root forms match the manual evaluator has marked it as True. But this is not the case for the automated evaluator. Although stemming has been done, but problems still persist which can be only solved by lemmatization, because Hindi is a highly inflected language.

5 Conclusion and Future Work

In this paper we present a completely new approach for extracting bilingual lexicon from comparable corpora. We show the results of experimentation on corpora of different levels of comparability. The basic feature of this approach is that it is language independent and needs no additional resource. We could not compare its performance with the Dependency Heterogeneity algorithm due to the lack of resources for Hindi. So this can be taken up as a future work. Also, the algorithm is quite inefficient. Another direction of research can be in trying to explore ways to reduce the complexity of this algorithm. We can also try to incorporate more linguistic information into this model instead of just word co-occurrence. It remains to be seen how these factors affect the performance of the algorithm. Another important question is what should be the size of the seed dictionary for optimum performance of the algorithm. This too can be taken up as a future research direction.

References

Dhillon, I., Y. Guan, and B. Kulis. 2007. Weighted graph cuts without eigenvectors: A multilevel approach. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 29:11:1944–1957, November.

Fung, Pascale and Lo Yuen Yee. 1998. An ir approach for translating new words from nonparallel, comparable texts. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics / the 17th International Conference on Computational Linguistics*, pages 414–420.

Fung, Pascale. 1995. Compiling bilingual lexicon entries from a non-parallel english-chinese corpus. In *Third Annual Workshop on Very Large Corpora*, Boston, Massachusetts, June.

Grefenstette, G. 1998. The problem of cross-language information retrieval. *Cross-language Information Retrieval*.

Kaji, H. 2003. Word sense acquisition from bilingual comparable corpora. In *Proc. of HLT-NAACL 2003 Main papers*, pages 32–39.

Kumano, T., H. Takana, and T. Tokunaga. 2007. Extracting phrasal alignments from comparable corpora by using joint probability smt model. In *Proc. of TMI*.

Och, F. and H. Ney. 2003. A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29(1):19–51, March.

Rapp, Reinhard. 1995. Identifying word translations in non-parallel texts. In *Proc. of TMI*.

Rapp, Reinhard. 1999. Automatic identification of word translations from unrelated english and german corpora. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics*, pages 519–526.

Saha, Sujan Kumar, Sudeshna Sarkar, and Pabitra Mitra. 1999. A hybrid feature set based maximum entropy hindi named entity recognition. In *Proceedings of the Third International Joint Conference on Natural Language Processing*, pages 343–349, Hyderabad, India, January.

Voorhees, E.M. 1999. The trec-8 question answering track report. In *Proceedings of the 8^{th} Text Retrieval Conference*.

Yuu, K. and J. Tsujii. 2009. Extracting bilingual dictionary from comparable corpora with dependency heterogeneity. In *Proc. of NAACL-HLT, short papers*, pages 121–124.

Zipf, George Kingsley. 1949. *Human Behaviour and the Principle of Least Effort: an Introduction to Human Ecology*. Addison-Wesley.