Phrase-Based Evaluation for Machine Translation

LI LiangYou   GONG ZhengXian[*] *ZHOU GuoDong*
School of Computer Science and Technology, Soochow University, Suzhou, China 215006
{20104227013,zhxgong,gdzhou}@suda.edu.cn

Abstract

This paper presents the utilization of chunk phrases to facilitate evaluation of machine translation. Since most of current researches on evaluation take great effects to evaluate translation quality on content relevance and readability, we further introduce high-level abstract information such as semantic similarity and topic model into this phrase-based evaluation metric. The proposed metric mainly involves three parts: calculating phrase similarity, determining weight to each phrase, and finding maximum similarity map. Experiments on MTC Part 2 (LDC2003T17) show our metric, compared with other popular metrics such as BLEU, MAXSIM and METEOR, achieves comparable correlation with human judgements at segment-level and significant higher correlation at document-level.

Title and Abstract in another language (Chinese)

基于短语的机器翻译自动评价

本文提出了基于短语的机器翻译自动评价方法, 并将高层次抽象信息加入到此方法中, 如语义相似度, 主题模型. 本文提出的方法主要有三个部分: 计算短语相似度, 为短语分配权重和寻找最大相似度匹配. 在MTC Part 2 (LDC2003T17)上的实验表明, 与BLEU, MAXSIM, METEOR等主流方法相比, 本文的方法与人工评价的相关性在句子级评价上取得了相当的效果, 在文档级评价上显著地高于其他方法.

Keywords: phrase similarity, topic model, machine translation, automatic evaluation.

Keywords in Chinese: 短语相似度, 主题模型, 机器翻译, 自动评价.

[*]Corresponding author.

1 Introduction

In recent years, machine translation (MT) has benefited a lot from the advancement of automatic evaluation which, compared with manual evaluation, can give quick and objective feedback on the quality of translation. So most of current MT systems need one or more automatic metrics to frequently update their models. Among all automatic evaluation metrics, those based on ngrams are most widely used. A basic mode of ngram-based metric is to estimate whether ngrams from system translation (also called **candidate**) can match with those from references or not. However, most of such metrics suffer from one or more following problems: 1) nonsense ngrams in evaluation; 2) same weight for different ngrams; 3) lack of fuzzy matching; 4) absence of context information.

Therefore, this paper proposes a new automatic MT evaluation metric which uses linguistic phrase rather than ngram as the basic unit of evaluation. In linguistics, a phrase is a group of words (or sometimes a single word) that form a constituent and so function as a single unit in the syntax of a sentence.[1] There are some different types of phrases, such as Noun Phrase (NP), Verb Phrase (VP), Adverb Phrase (ADVP), Adjective Phrase (ADJP), and Preposition Phrase (PP) and so forth. In this paper, only NP and VP are used in our experiments, and all phrases are obtained by chunker[2].

Given phrases, our metric evaluates translations with three key parts, including calculating phrase similarity, allocating weight to each phrase, and finding a maximum similarity map. For the first part, we not only adopt a semantic similarity function based on WordNet but also explore a topic similarity function based on a popular topic model. And we present a novel framework to unify the two similarity measures successfully. To the second part, we examine several different weight functions, including phrase length (i.e. ngram weight), tf.idf and topic relevance to distinguish informativeness of phrases. To the last part, we address how to establish a maximum similarity map between phrases of candidates and references and further analyze its working mechanism by experiments.

It is worth to mention that our metric has a great flexibility such that any other similarity and weight functions could be incorporated easily. Experiments also show the metric, compared with some popular metrics, achieves comparable correlation with human judgements at segment-level and significant higher correlation at document-level.

2 Related works

In recent years, numerous ngram-based metrics have been proposed. BLEU (Papineni et al., 2002) as the most famous evaluation metric calculates an overall score via geometric mean of precisions on different ngrams. NIST (Doddington, 2002) improves BLEU with arithmetic mean and weight for different ngrams. However both BLEU and NIST do not consider synonyms. In METEOR (Banerjee and Lavie, 2005), three modules, “exact”, “porter stem” and “WN synonymy”, are used to create word-alignment successively. And a penalty for word-order is integrated into the final score. MAXSIM (Chan and Ng, 2008) constructs a bipartite graph for unmatched ngrams. And Kuhn-Munkres algorithm (Kuhn, 1955; Munkres, 1957) is used to find a maximum weighted matching. However, synonyms in METEOR and MAXSIM are viewed as equivalent completely. Furthermore, nonsense ngrams are still used in these metrics. By contrast, in our metric, phrases are considered as the unit of evaluation and a fine similarity

[1] http://en.wikipedia.org/wiki/Phrase

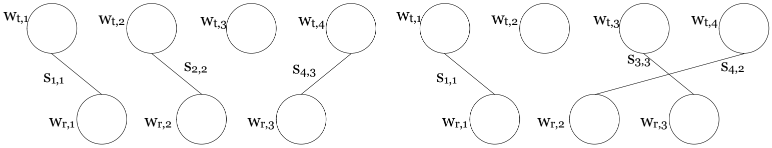
[2] http://jtextpro.sourceforge.net/

function is defined. And context information contributes to our metric as well.

Phrase information has been used in several evaluation methods. In the work of Giménez and Màrquez (2007), overlapping is calculated on the set of words within a same phrase type, and sequences of phrase types are used in the metric of NIST to score phrase-order. However, this work does not distinguish different phrases with the same type and ignores the fact that different types of phrases can be established a correspondence. Echizen-ya and Araki (2010) propose to establish correspondence of phrases for which mutual similarity score is highest. But this method just takes NP into consideration since similarity based on PER (Su et al., 1992) cannot determine the correspondence of VP correctly. Zhou et al. (2008) diagnoses translations based on check-points where each phrase can be scored by ngram matching. However, it ignores the order of phrases in a translation and phrase correspondence relies on word-alignment trained on parallel corpus. Different with these works, this paper treats a phrase as a single unit and integrates explicit measurement of phrase-order into metric and correspondence is established by fine similarities between phrases.

3 Phrase-based evaluation metric

Phrase-based evaluation (PBE) metric proposed by this paper compares a pair of candidate-reference translation by identifying phrase correspondence between them. Firstly, this metric extracts phrases from them using chunking tool; then each phrase is assigned a weight to indicate its informativeness. After that, according to similarities between phrases, the metric find a maximum similarity map between two phrase sequences so that each phrase of one translation is correspondent with at most one in the other. Figure 1 gives two examples of mapping.



(a) Map with 2 blocks

(b) Map with 3 blocks

Figure 1: Examples of mapping. w is the weight of a phrase in candidate t or reference r; and s is the similarity between two phrases

Given a maximum map, we can calculate precision scores for candidate t and reference r respectively and a penalty factor, similar to Banerjee and Lavie (2005), to measure phrase-order:

$$P_t = \sum_{i \in t} w_i s_i / \sum_{i \in t} w_i \qquad P_r = \sum_{j \in r} w_j s_j / \sum_{j \in r} w_j \qquad pen = \gamma \left(\#blocks - 1/m\right)^{\beta}$$

where w_i is the weight of the ith phrase and s_i is the similarity related to this phrase, γ and β are constant, m and $\#blocks$ are the number of matchings and blocks[3] in the map. Then final score for evaluation is:

$$score = \left(\alpha P_t + (1-\alpha) P_r\right) \cdot \left(1 - pen\right)$$

[3] A block in a map consists of only as more consecutive matchings as possible. For example, in Figure 1(a), there are two blocks: one consists of $s_{1,1}$ and $s_{2,2}$; $s_{4,3}$ is the other one; Figure 1(b) has three blocks.

where α is a constant varying from zero to one.

In this paper, similar to METEOR, document-level score is obtained by integrating fragments of scores from segments. However, in our metric, context information is also used to improve the performance of document-level evaluation.

4 Phrase similarity

In this paper, phrase similarity consists of two parts and can be represented by an interpolation:

$$SIM = \theta SIM_{in} + (1-\theta) SIM_{ctxt}$$

where θ is a constant ranging from zero to one; SIM_{in}, called internal or general function, is only related to phrases and separated from their contexts; external SIM_{ctxt} is also called context similarity which is circumstance-specific.

In this paper, internal similarity is based on WordNet, defined as SIM_{WN} and external similarity is measured by topic similarity SIM_t. The rest of this section will describe the two functions.

4.1 Similarity function based on WordNet

Ignoring word-order, each pair of words between two phrases can have a lexical similarity. So similarity between phrases can be measured by similarities of words.

Lexical Similarity: Given two words, their similarity is one if they have the same lemma or porter stem. Otherwise, WordNet is used to compute a semantic similarity. However, different with other metrics where similarity of any two synonyms is one, our metric uses a fine function proposed by Lin (1998)[4]:

$$lin\left(c_1, c_2\right) = 2\log P\left(c_0\right) / \left[\log P\left(c_1\right) + \log P\left(c_2\right)\right]$$

where c_1 and c_2 are two synsets, c_0 is the lowest level of synset which subsumes c_1 and c_2, $P(c)$ is the probability of a word belonging to synset c.

Similarity between two phrases: For two phrases $phr_1 = w_1 w_2 \cdots w_m$ and $phr_2 = v_1 v_2 \cdots v_n$, there would be in total $m \times n$ lexical similarities. According to Liu et al. (2008), these similarities can be presented in a matrix where the element at position of $\left(i, j\right)$ corresponds to the value of lexical similarity $sim\left(w_i, v_j\right)$. Then, a similarity between the two phrases can be obtained by:

$$SIM_{wn}\left(phr_1, phr_2\right) = \left[S\left(phr_1, phr_2\right) + S\left(phr_2, phr_1\right)\right] /2$$

where $S\left(phr_1, phr_2\right) = \sum_{i=1}^{m} \max\{sim\left(w_i, v_1\right) \cdots sim\left(w_i, v_n\right)\}/m$.

4.2 Similarity function based on topic model

There are three steps to use topic model as external similarity for phrases: topic model estimation, obtaining topic distributions of phrases and calculating topic similarity of phrases.

Topic model estimation: In this paper, We use Latent Dirichlet Allocation[5] (LDA) (Blei et al., 2003), an unsupervised machine learning technique, to obtain topic model from data collection.

[4] http://www.sussex.ac.uk/Users/drh21/

[5] http://jgibblda.sourceforge.net/

This model could give two type of distributions: $p(w \mid z)$ and $p(z \mid d)$. It is worth to mention that we build separate topic models for references and candidate translations of each system. This is because separate models can prevent unknown or semantically equivalent words between different systems from being underestimated.

Topic distributions of phrases: For a phrase $phr = w_1 \cdots w_n$, its probability on topic z can be calculated as follows:

$$\begin{aligned}P\left(topic = z \mid phr = w_1 \cdots w_n\right) &= P\left(w_1 \cdots w_n \mid z\right) \cdot P(z) / P\left(w_1 \cdots w_n\right) \\ &= \prod_{i=1}^{n} P\left(w_i \mid z\right) \cdot P(z) / \prod_{i=1}^{n} P\left(w_i\right) \\ &= \prod_{i=1}^{n} P\left(w_i \mid z\right) \cdot P(z) / \prod_{i=1}^{n} \sum_{k=1}^{K} P\left(w_i \mid Z_k\right) P\left(Z_k\right)\end{aligned}$$

where K is the number of topics and Z is the set of topics. In this paper, $p(z) = 1/K$. Note that this equation treats phrase as bag of words and the same phrases in different documents within a topic model have the same topic distribution. Thus such distribution is topic-specific rather than document-specific.

Topic similarity of phrases: Generally, similarity between topic distributions of two phrases can be calculated by cosine function. However, in this paper, topic is not aligned between different models and thus two distributions from different models cannot be used in cosine function directly. In this paper, we adopt a simplified method. Given two phrases phr_t and phr_r from document d_t of one system and d_r of one reference, their topic similarity is:

$$SIM_t\left(phr_t, phr_r\right) = 1 - \mid cos\left(phr_r, d_r\right) - cos\left(phr_t, d_t\right) \mid$$

where $cos\left(phr, d\right)$ denotes cosine value between topic distributions of phrase phr and document d. This Equation suggests that if two phrases have approximate phrase-document similarity, so does their mutual similarity. Such topic similarity is document-specific. Of course, it is possible that two translations differ too much while two phrases in them get a higher final similarity. However, our experiment shows that a bias to internal similarity can reduce such influence.

5 Phrase weight

In this section, we will present two basic functions: ngram, tf.idf. Then a method of improving them with topic model is described.

5.1 Basic weight functions

In our metric, **ngram**, length of a phrase, is the default weight function. However, this function ignores the contexts of a phrase. Thus another function, **tf.idf** which has been used in information retrieval widely, is also presented:

$$tf.idf_{t,d} = (1 + \log(tf_{t,d})) \log\left(N/df_t\right)$$

where $tf_{t,d}$ is the number of occurrence of the term t in the document d, df_t is the number of documents which contains term t, N is the number of documents.

It is worth noting that in this paper, we build their own tf.idf dictionaries for references and candidates of each system. This is different from other works, such as Babych and Hartley

(2004) and Wong and Kit (2009), where tf.idf value of a word in candidate is directly taken from references and thus unmatched words in the candidate are ignored.

5.2 Topic-based weight

With close scrutiny, phrase weight can be divided into two parts:

$$Weight\left(phr,t\right) = Rel\left(phr,t\right) \cdot Info\left(phr\right) \qquad (1)$$

where $Info\left(phr\right)$ denotes informativeness of phrase phr, $Rel\left(phr,t\right)$ measures the correlation between the phrase and its text t. Ideally, we expect the function $Info\left(phr\right)$ has little correlation with t.

In general, we could measure the correlation between phr and t from different perspectives, such as topic relevance, probability of co-occurring and so on. In this paper, we define: $Rel\left(phr,t\right) = cos\left(phr,t\right)$. And functions in section 5.1 can serve as $Info$. However, It should be noted that tf.idf value of a word or phrase relys on its contexts to some extent.

6 Maximum similarity map

Similar to Chan and Ng (2008), we view matching between phrases as a bipartite graph and Kuhn-Munkres (KM) algorithm is use to find a map which has a maximum sum of similarities. However, our metric needs to calculate a penalty score for phrase-order. Thus when there are multiple such maps, we need to select one from them.

In this paper, facing with multi-options, KM algorithm will select the maps in which the first phrase of current reference has the minimal correspondent position in candidate; and this process will continue in the phrase sequence of the reference until there's only one map left. This stratagem would keep the relative order of phrases in some situations which will be illustrated in section 7.3.

Take Figure 1 as an example. KM will choose Figure 1(a), because in both maps $w_{r,1}$ has the same correspondence $w_{t,1}$ while $w_{r,2}$ has the correspondence $w_{t,2}$ in Figure 1(a) and $w_{t,4}$ in Figure 1(b).

7 Experiments

We conduct experiments on MTC Part 2 (LDC2003T17) which contains 100 source documents (878 segments in total) in Chinese and 4 English references for each segment. Translations of three systems were assessed by human judges on each segments in terms of adequacy (Adq) and fluency (Flu). We normalize the human raw scores according to Blatz et al. (2004) and average scores for segments. Document score is the average of scores of its segments. Before evaluation, translations are tokenized and lower-cased.

In our default metric PBE (or PBE$_{\text{ngram}}$), α is set to 0.2, both β and γ are 0.5, θ is 1 and phrase weight function is ngram. In this paper, only NP and VP are taken into consideration since they contain more information and give a stable evaluation in our preliminary experiments. Pearson correlation coefficient is used to measure correlation between automatic evaluation and human judgements.

7.1 Performance of default metric

According to Table 1[6], our metric PBE is significantly better than other three popular metrics only with an exception on METEOR[7] at segment-level. We guess the reason of relative lower performance of our metric at segment-level than document-level is that short segments do not contain enough phrases and thus PBE can not perform well on them. Furthermore, Table 1 shows tf.idf brings the best metric $\text{PBE}_{\text{tfidf}}$, suggesting that context information can help to improve evaluation effectively. In addition, since these metrics put more effort on matching between candidates and references, they are more correlated with Adq than Flu score.

Metric	Segment-Level		Document-Level	
	Adq	Flu	Adq	Flu
BLEU	0.2379	0.2184	–	–
MAXSIM	0.2677	0.2235	0.2722	0.2600
METEOR	0.3489	0.3014	0.3025	0.2938
PBE	0.3262	0.3199	0.3807	0.3291
$\text{PBE}_{\text{tfidf}}$	–	–	0.4153	0.3471

Table 1: Pearson coefficient for automatic evaluation metrics

7.2 Effect of topic model in evaluation

Table 2 is the result of evaluation based on topic model.[8] We can find that topic model can improve metrics significantly. An exception happens on $\text{PBE}_{\text{tfidf}}$: topic-based weight function "Weight" seems helpless. We guess this results from the potential relevance between tf.idf and our topic model: both rely on context information within a document and corpus.

Metric	Topic-Based Func.	Document-Level	
		Adq	Flu
$\text{PBE}_{\text{ngram}}$		0.3807	0.3291
	+Weight	0.4065	0.3503
	+SIM	0.4007	0.3380
	+Weight+SIM	0.4285	0.3648
$\text{PBE}_{\text{tfidf}}$		0.4153	0.3471
	+Weight	0.4176	0.3439
	+SIM	0.4428	0.3626
	+Weight+SIM	0.4324	0.3519

Table 2: Pearson coefficient for metrics based on topic model with K=50 and θ=0.8

[6] Tf.idf is tested only on document since document is more suitable for it to make estimation for phrase weight. And we do not report performance of BLEU at document-level because it's unfair to compare it with other metrics since BLEU considers impact of sentence length.

[7] METEOR (Denkowski and Lavie, 2011) uses parameters tuned to adequacy scores.

[8] LDA is trained on documents, thus only results at document-level evaluation are presented. And preliminary experiments suggest that metrics based on topic model perform better when K=50 and θ=0.8. Thus this setting is also used in this paper.

7.3 Selection of maximum similarity map

For comparing with our selection strategy for multi-options, we use beam search to find a “better” map which has less blocks without changing the maximum similarity. Our experiment shows that there is only one maximum similarity map in most cases; otherwise, in most situations of multi-options, our strategy will give a better and reasonable results.

For example, in Figure 2, each translation has two “peace” and beam search finds a different map with KM where the number of blocks declines by 1. However, this result seems to lead to an overestimation since it destroys the original order of “peace” for the sake of lower *pen* value (see related equation in section 3).

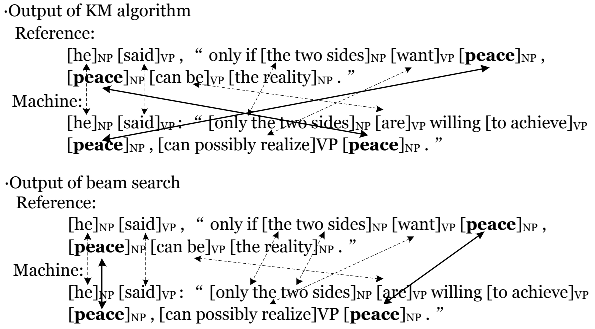

·Output of KM algorithm
Reference:
[he]NP [said]VP , “ only if [the two sides]NP [want]VP [peace]NP ,
[peace]NP [can be]VP [the reality]NP . ”
Machine:
[he]NP [said]VP : “ [only the two sides]NP [are]VP willing [to achieve]VP
[peace]NP , [can possibly realize]VP [peace]NP . ”
·Output of beam search
Reference:
[he]NP [said]VP , “ only if [the two sides]NP [want]VP [peace]NP ,
[peace]NP [can be]VP [the reality]NP . ”
Machine:
[he]NP [said]VP : “ [only the two sides]NP [are]VP willing [to achieve]VP
[peace]NP , [can possibly realize]VP [peace]NP . ”

Figure 2: An example of comparison between results of KM and beam search

Conclusion and Future Work

This paper present a new automatic MT evaluation metric which is based on linguistic phrase. This metric incorporates high-level abstract information such as semantic similarity based on WordNet and topic model into phrase similarity and explores several functions such as ngram, tf.idf and topic relevance to allocate weight for each phrase. And a method of finding a maximum similarity map is presented. Experiments show our metric is more suitable for long translation and achieves significant higher correlation with human judgements than several other popular metrics at document-level and comparable results at segment-level. Experimental results also show that context information and topic model can improve the performance of evaluation effectively.

In the future, we would examine in details how chunker performs on translations with various qualities, use syntactic information or structure in evaluation and explore utilization of more sophisticated model instead of bag-of-word etc. We expect our metric could be performed on document-level SMT systems (Gong et al., 2011) to measure their quality rightly.

Acknowledgments

This research is supported by the National Natural Science Foundation of China under grant No.61273320 and 61003155, the National High Technology Research and Development Program of China (863 Program) under grant No.2012AA011102, and the Preliminary Research Project of Soochow University.

References

Babych, B. and Hartley, A. (2004). Extending the BLEU MT evaluation method with frequency weightings. In *Proceedings of ACL 2004*. Association for Computational Linguistics.

Banerjee, S. and Lavie, A. (2005). METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings of the ACL 2005 Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pages 65–72. Association for Computational Linguistics.

Blatz, J., Fitzgerald, E., Foster, G., Gandrabur, S., Goutte, C., Kulesza, A., Sanchis, A., and Ueffing, N. (2004). Confidence estimation for machine translation. In *Proceedings of COLING 2004*. Association for Computational Linguistics.

Blei, D. M., Ng, A. Y., and Jordan, M. I. (2003). Latent dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022.

Chan, Y. S. and Ng, H. T. (2008). MAXSIM: A maximum similarity metric for machine translation evaluation. In *Proceedings of ACL 2008: HLT*, pages 55–62. Association for Computational Linguistics.

Denkowski, M. and Lavie, A. (2011). Meteor 1.3: automatic metric for reliable optimization and evaluation of machine translation systems. In *Proceedings of WMT 2011*, pages 85–91. Association for Computational Linguistics.

Doddington, G. (2002). Automatic evaluation of machine translation quality using n-gram co-occurrence statistics. In *Proceedings of HLT 2002*, pages 138–145. Morgan Kaufmann Publishers Inc.

Echizen-ya, H. and Araki, K. (2010). Automatic evaluation method for machine translation using noun-phrase chunking. In *Proceedings of ACL 2010*, pages 108–117. Association for Computational Linguistics.

Giménez, J. and Màrquez, L. (2007). Linguistic features for automatic evaluation of heterogenous MT systems. In *Proceedings of WMT 2007*, pages 256–264. Association for Computational Linguistics.

Gong, Z., Zhang, M., and Zhou, G. (2011). Cache-based document-level statistical machine translation. In *Proceedings of EMNLP 2011*, pages 909–919. Association for Computational Linguistics.

Kuhn, H. W. (1955). The Hungarian method for the assignment problem. *Naval research logistics quarterly*, 2:83–97.

Lin, D. (1998). An information-theoretic definition of similarity. In *Proceedings of ICML 1998*, pages 296–304. Morgan Kaufmann Publishers Inc.

Liu, Y., Li, C., Zhang, P., and Xiong, Z. (2008). A query expansion algorithm based on phrases semantic similarity. In *Proceedings of ISIP 2008*, pages 31–35. IEEE Computer Society.

Munkres, J. (1957). Algorithms for the Assignment and Transportation Problems. *Journal of the Society for Industrial and Applied Mathematics*, 5:32–38.

Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. (2002). BLEU: a method for automatic evaluation of machine translation. In *Proceedings of ACL 2002*, pages 311–318. Association for Computational Linguistics.

Su, K.-Y., Wu, M.-W., and Chang, J.-S. (1992). A new quantitative quality measure for machine translation systems. In *Proceedings of COLING 1992,* pages 433–439. Association for Computational Linguistics.

Wong, B. and Kit, C. (2009). ATEC: automatic evaluation of machine translation via word choice and word order. *Machine Translation*, 23:141–155.

Zhou, M., Wang, B., Liu, S., Li, M., Zhang, D., and Zhao, T. (2008). Diagnostic evaluation of machine translation systems using automatically constructed linguistic check-points. In *Proceedings of COLING 2008*, pages 1121–1128. Association for Computational Linguistics.