Query Expansion for Mining Translation Knowledge from Comparable Data

Lu Xiang[†] Yu Zhou[†] Jie Hao[‡] and Dakun Zhang[‡]

[†]NLPR, Institute of Automation Chinese Academy of Sciences, Beijing, China
[‡]Toshiba (China) R&D Center
[†]{lu.xiang, yzhou}@nlpr.ia.ac.cn
[‡]{haojie, zhangdakun}@toshiba.com.cn

Abstract. When mining parallel text from comparable corpora, we confront vast search space since parallel sentence or sub-sentential fragments can be scattered throughout the source and target corpus. To reduce the search space, most previous approaches have tried to use heuristics to mine comparable documents. However, these heuristics are only available in few cases. Instead, we go on a different direction and adopt the cross-language information retrieval (CLIR) framework to find translation candidates directly at sentence level from comparable corpus. What's more, for the sake of better retrieval result, two simple but effective query expansion methods are proposed. Experimental results show that using our query expansion methods can help to improve the recall significantly and obtain candidates of sentence pairs with high quality. Thus, our methods can help to make good preparation for extracting both parallel sentences and fragments subsequently.

Keywords: comparable data, parallel text mining, statistical machine translation, cross-language information retrieval, query expansion

1 Introduction

The parallel corpora are resources of great importance for many natural language processing tasks, especially for statistical machine translation (SMT), while parallel corpora with high quality is expensive. In order to alleviate the lack of parallel data, many researchers have turned to mine the large amount of available comparable data on Internet. In mining parallel text from comparable data, a great challenge is that the search space is quite vast, which makes it difficult to obtain good parallel resource and the process is too slow to be applied in the practical application.

To reduce the search space, much work [2, 8, 9, 11, 13, 14] utilizes some heuristic information in which they step document alignment first and then only inspect the sentences in the aligned document pairs. However, in many situations, the simple heuristic information such as URL and title are not available which makes document alignment hard to actualize.

In this paper our purpose is to reduce the search space without document-level alignment. We believe it will be much helpful to narrow the search space if we can pre-select part of comparable sentence pairs that most possibly contain parallel text because the size of comparable data is huge. Inspired by this motivation, we resort to the cross-language information retrieval (CLIR) framework to pre-filter the candidates, which indexes the target corpus directly at sentence-level and adopts a search engine to find the sentences in target corpus that are the most likely translations given a source sentence. The best CLIR-returned sentences are kept as candidates.

Some research work [10, 15] has been done on how to use CLIR method for selecting candidate sentences. Among these work, Ştefǎnescu et al. [15] uses the dictionary-based CLIR framework to select candidates. However, a word usually has several translations, for example, the word "reaction" has 6 meanings as shown in Table 1, and some translations are not right in some specific contexts. We believe it will add extraneous terms to the query and thus will degrade the quality of candidate sentence collection if we just simply use all the dictionary translations. This will definitely affect the parallel text extraction procedure afterwards.

Table 1. An example of ambiguities in a dictionary

Terms	**Translation Candidates**
reaction	反应; 感应; 反动; 复古; 反作用; 影响
increase	增加; 增长; 提高; 繁殖; 扩大; 增添
intensify	加剧; 加强; 强化; 增强; 神话

Based on the analysis above, we know that it is inappropriate to use CLIR without any modification. Therefore, this paper presents two query expansion methods for the translation of queries from source language into target language. One is word-level translation and the other one is phrase-level translation. In word-level translation, a bilingual dictionary is utilized to translate the content words in the source sentence. Unlike the work of Ştefǎnescu et al. [15] that uses all the dictionary translations, we propose a word disambiguation algorithm using beam-search that only uses the monolingual target corpus to select the better word sequence to form the query. In phrase-level translation, a simplified translation model which only uses phrase and lexical translation probability is proposed for the translation. Experiments show that our proposed query expansion methods can not only help to reduce the space efficiently, but also can achieve a better collection of candidate sentence pairs.

The remainder of this paper is organized as follows: Section 2 introduces the related work. Section 3 gives the CLIR framework for candidate sentence generation and describes our two query expansion methods in detail. Section 4 presents the experiments. Finally, we conclude the paper in Section 5.

2 Related work

Mining parallel text from comparable data has attracted many researchers and much research work has been done on this task. However, it presents many difficulties and one of the greatest obstacles is the vast search space. Much work has been done to prune the search space and all these methods can be classified into two categories: (1) document level pre-filtering, and (2) sentence level pre-filtering.

The general way for document level pre-filtering is to perform document alignment first and then to inspect the sentences in the aligned document pairs only. This road has been taken by many researchers. [2, 3] adopt a bilingual dictionary to compute document similarity for document alignment. [8] uses dictionary-based cross-lingual information retrieval method to get more precise article pairs. [13] implements a hash-based algorithm to directly compute the cross-lingual pairwise similarity to find article pairs. All these methods need to calculate pairwise similarities across the huge bilingual corpora and it's quite computationally intensive. To reduce the computational complexity, most studies fall back to heuristics. [8] just compares news articles published close in time. [11] exploits "inter-wiki" links in Wikipedia to align documents. After document alignment, we can mine parallel resources from the aligned documents.

Since high-quality document-level alignment is difficult to acquire in many situations, some work has tried to pre-select candidate sentence pairs at sentence-level. [12] adopts a beam-search algorithm to extract parallel sentences directly at sentence-level without document alignment. [5, 10] employ a SMT system to translate the source part of comparable corpus and then use the translations as queries to conduct information retrieval to find candidate sentences. However, there are not enough resources between many language pairs to build a SMT system. [15] uses the dictionary-based CLIR framework to generate candidates. But as mentioned before, the serious ambiguity problem existed in a dictionary will affect the performance seriously.

3 Candidate Sentence Generation from Comparable Data

The pipeline of using CLIR framework to generate candidate sentence pairs is shown in Fig. 1. From the figure we can see, all of our processing is directly at the sentence-level. The procedure can be divided into three steps: (1) Building index for the target corpus; (2) Translating the source sentence into target language to form queries; (3) Search for candidate sentence pairs.

3.1 Indexing Target Sentences

To implement our framework, we need to build index for the target corpus first. Splitting the target corpus into sentences and performing a series of basic operations that are similar with the process of Chinese word segmentation or English tokenization and stemming. We use the Java implementation of Lucene[1] to index the target sentences as Lucene documents.

We also compute the length of each sentence. We believe the length information can help us to filter out some non-parallel sentences further because the ratio of the lengths of two sentences that are translations of each other must be within a certain range. Therefore, for each Lucene document, we introduce the following two searchable fields:

[1] http://lucene.apache.org/

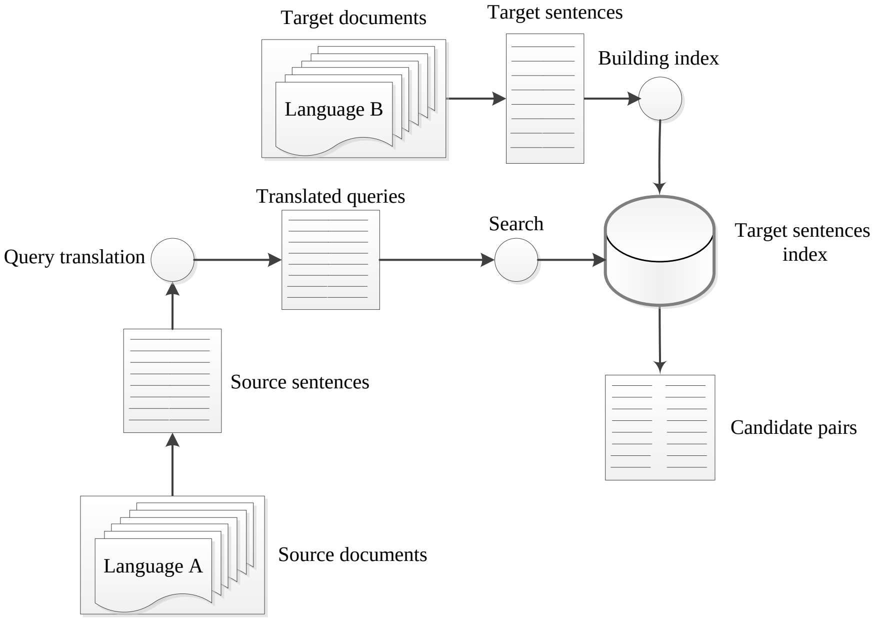

Fig. 1. Our framework to generate candidate sentence pairs

(a) A field storing the target sentence;
(b) A field storing the length of the sentence.

Then we build full-text indexing on these two fields for target sentences. The index structure is extensible and we can add other useful field information easily.

3.2 Generating Query from Source Sentence

After finishing indexing target sentences, our next step is considering how to translate source sentences into target language to generate queries. In our model, for word-level transformation, we adopt a machine-readable bilingual dictionary to translate source sentence into target word by word. And to solve the ambiguity problem mentioned before, we present a beam-search algorithm using nothing more than the monolingual target corpus to select the best translation sequence. For phrase-level transformation, we use a small collection of bilingual corpus to train a simplified translation model for the translation.

3.2.1 Word-level Transformation

To map the information from source sentences into target, the most simple and convenient way is to utilize the bilingual machine readable dictionaries. However, the dictionary translations are usually ambiguous and thus will affect the retrieval results.

Much efforts have been done to solve the problem of disambiguation [4, 6, 7] and most of the existing approaches exploit the word co-occurrence patterns and then use a greedy algorithm to select an optimal translation set.

Instead, we deal with this problem from a different aspect. First, we see the co-occurrence of possible translation terms as a graph and the Mutual Information (MI) values are the weight between two words. Fig. 2 gives an example of such a graph. The square nodes under w_1, w_2 and w_3 represent the translations of the three words respectively and the links indicate the MI values are available for the pairs.

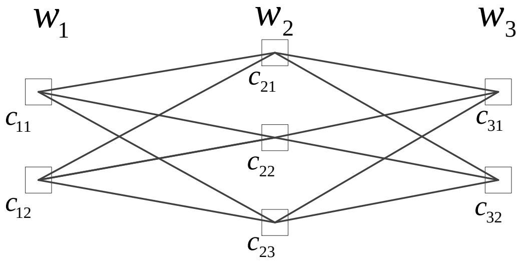

Fig. 2. An example of word co-occurrence graph

MI can be used to evaluate the significance of word co-occurrence associations and is defined as the following formula (1) [1]:

$$\mathrm{MI}(x,y)=\log_2\frac{p(x,y)}{p(x)\,p(y)} \tag{1}$$

Here, $p(x,y)$ is the co-occurrence probability of x and y within a window size[2]. $p(x)$, $p(y)$ and $p(x,y)$ are the maximum likelihood estimates of the corresponding probabilities.

Given a source sentence S with a set of n content words $\{e_1,e_2,\cdots,e_n\}$ and a set of translation T obtained from the bilingual dictionary, $T(e_i)=\{c_{i1},c_{i2},\cdots,c_{im}\}$, we need to select the best translation for each content word to form the final translation set. However, to find such an optimal set will cost large amount of computation. Instead, we treat the problem as the optimal path selection and use a beam search algorithm to search for the B-best path.

Here, we define the path as a set of words $\{t_1,t_2,\cdots,t_n\}$, where $t_i\in T(e_i)$.

For example, in Fig. 2, w_1 has two translations, denoted as $T(w_1)=\{c_{11},c_{12}\}$. Similarly, we have $T(w_2)=\{c_{21},c_{22},c_{23}\}$ and $T(w_3)=\{c_{31},c_{32}\}$. It is easy to see $\{c_{11},c_{21},c_{31}\}$ and $\{c_{12},c_{21},c_{31}\}$ are possible two paths.

[2] In our application, $\mathrm{p}(x,y)$ is the co-occurrence probability of x and y within a sentence.

Our optimal algorithm to select path relies on the MI values. The MI values are based on the assumption that the words co-occur in the same query, and their proper corresponding translations are likely to co-occur in the same documents. We think the MI values can reflect the semantic association between words in some degree. On the basis of this idea, our evaluation function of $\text{Path}\left(t_1,t_2,\cdots,t_n\right)$ is shown below (2):

$$\text{Score}\left(\text{Path}\right)=\sum_{t_i,t_j\in Path,i\neq j} MI\left(t_i,t_j\right) \tag{2}$$

The best Path $Path^*$ is defined below (3):

$$Path^*=\arg\max\left(Score\left(Path\right)\right) \tag{3}$$

We use the beam-search algorithm for the best word translation sequence with the highest path score. The algorithm is given in Fig. 3, where T is a set of translations as mentioned before, and the variable *candidate* represents one of the current path kept in the *candidates*. *T'* is the translation candidates of one word and *c''* is one translation in *T'*. The *agenda* is a sequential list, used to keep all the paths generated at each stage, ordered by the score which is calculated using formula (2). The variable *candidates* is the set of paths that can be used to generate new path, that is the *B*-best path from previous stage. And *B* is the number of paths retrained at each stage. Its value is an important factor to the performance of the algorithm and we set *B* to 128 in our application.

```
function Beam-Search( T,agenda,candidates, B)
    candidates ← T($e_1$)
    agenda ← CLEAR(agenda)
    for T' in { T($e_2$), T($e_3$), ⋯, T($e_n$)}:
        for candidate in candidates:
            for c'' in T':
                agenda ←ADD(candidate, c'')
        candidates ← TOP-B(agenda,B)
        agenda ← CLEAR(agenda)
```

Fig. 3 The beam-search algorithm

CLEAR empties the agenda and removes all the items from agenda. ADD refers to an operation that add a new translation node to expand the path and TOP-B returns the highest B scoring paths from the agenda.

3.2.2 Phrase-level Transformation

In word-level transformation, the basic unit of translation is a single word. Though our proposed beam-search method to select word sequence can help to discard some error translations, the improvement of performance is limited since we only use the co-occurrence information in the target corpus and the context of the word in the source sentence is not available. However, the context of the word to be translated is helpful for the selection of translation. Thus, we can enlarge the translation unit from word-level to phrase-level to use the context information.

At present, phrase-based SMT model usually consists of three factors: the phrase translation table, the reordering model, and the language model. But in fact, even if the sentence is not smooth or the order of the words is not proper, we can retrieve a satisfied result set if the words given to the search engine are correct. In order to reduce the computational complexity, we propose to use a simplified phrase translation model that only uses the phrase translation table to translate source sentence into target language. Here, four features are used to choose the translation: phrase translation probability $\varphi(f|e)$, $\varphi(e|f)$ and lexical weighting $\mathrm{lex}(f|e)$, $\mathrm{lex}(e|f)$. A small bilingual parallel corpus is used to train the translation model.

3.3 Searching for Candidate Sentences

After translating the source sentence into target language, we can obtain a set of target words. A boolean model is used to retrieve the documents and each of the target words is added as a disjunctive query term (the OR operator). In order to narrow the search space and obtain the better result, a range filter is built. This kind of filter makes the search engine only search the subset of sentences in which the ratio of sentence and the source sentence is within a certain range. In our experiments, the range is set as [0.5, 2]. After the query is constructed, we use it to feed the Lucene search engine to get the best *h* hits.

4 Experiments

4.1 Experiments setup

We want to measure the performance of using our query translation methods for the search engine to find translation candidates. We conduct our experiments on manually created English-Chinese data set. The target corpus, denoted as FBIS&NIST, consists of the Chinese side of FBIS corpus, NIST MT 2003 and NIST MT 2005. We use Lucene to index FBIS&NIST. The English side of NIST MT 2003 (denoted as EN-03) and NIST MT 2005 (denoted as EN-05) are used as the source corpus. Table 2 shows the relevant statistics of the target corpus and source corpus.

Table 2. Statistics of the target and source corpus

	sentences
FBIS&NIST	237,671
EN-03	919
EN-05	1082

Each sentence in the source corpus is used as query and the best *k* hits (target sentences) returned by the search engine are kept as candidates. Ideally, we would like to see the real translations in the candidates so that we can have a chance to extract them in the next steps. To measure the quality of retrieval results, we define recall as follows: Let *G* denote the total sentence number of the source corpus. Each sentence will obtain a set of target sentences as translation candidates and then we will have *G* such retrieval sets. Let *R* denote the number of retrieval sets that contain the real translations. Then

$$\mathrm{Recall} = \frac{R}{G} * 100 \tag{4}$$

The higher recall means more retrieval sets contain the real parallel text which is very important to the following steps.

4.2 Evaluation

We first design experiments to evaluate the efficiency of our proposed two query translation methods. In word-level translation, we use a common English-Chinese dictionary containing 41,814 entries. In phrase-level translation, we adopt the Moses toolkit and FBIS corpus with 235,670 sentence pairs to train the SMT system.

Results are shown in Table 3. Here, in baseline system [15], we use the bilingual dictionary to translate the source sentence and using all the translations as query terms to search for target sentences. Method_1 refers to use the beam-search word sequence selection method and Method_2 refers to use the simplified translation model described in sub-section 3.2.1 and sub-section 3.2.2 respectively.

Table 3．Results of using different query translation method（%）

The number of *k*		**1**	**5**	**10**	**20**	**50**
EN-03	Baseline	59.19	73.12	76.82	81.39	84.54
	Method_1	71.38	79.86	84.76	87.59	89.44
	Method_2	72.79	82.91	86.50	88.46	90.75
EN-05	Baseline	54.89	70.14	73.84	79.02	84.75
	Method_1	65.71	76.61	82.71	85.85	89.27
	Method_2	74.86	84.47	86.78	88.81	91.49

As Table 3 shows, the recall increases with the *k* increasing. Both on EN-03 and EN-05, our proposed two query translation methods improve the performance significantly compared to the baseline, which proves the effectiveness of our methods. On one hand, our methods can help to achieve a much better quality of candidates which is important to the following parallel text mining steps. On the other hand, even if we only keep the best 10 hits but the baseline keeps the best 50 hits for each sentence, we can obtain the comparable recall. This means the search space can be reduced about 10 times.

We can see that the performance of using the simplified translation model is better than that of using the beam-search word sequence selection method. This is mainly because the beam-search method to select word sequence only uses the co-occurrence information in the target corpus while the translation model uses the context of the word to be translated.

As we know the performance of the translation model is domain-sensitive. In order to make a more comprehensive comparison of every method, we also conduct a cross-domain experiment. We have 1,500 sentence pairs from computer science domain and add the Chinese side into the former built index. The English side, denoted as COM, is used as source corpus to retrieve candidates. The results are shown in Table 4.

Table 4． Cross-domain evaluation result（%）

the number of *k*		**1**	**5**	**10**	**20**	**50**
COM	Baseline	57.33	71.40	75.40	79.60	85.20
	Method_1	67.66	78.73	82.6	84.66	88.33
	Method_2	35.93	48.80	53.53	58.13	65.20

From Table 4 we can see that our beam-search method improves the performance significantly while the performance of using the translation model decreases a lot. This is because the bilingual dictionary we used is a general dictionary. We believe that the performance can be further promoted if we enlarge the dictionary. However, the translation model is trained on a news domain data and it performs poorly on the other domain which declines the retrieval performance.

Comparing the performance of using different query translation method shown in Table 3 and Table 4, we can conclude that both our proposed beam-search method to select word sequence and the simplified translation model are very helpful to transform source sentence into target language to obtain a better quality of candidate sentence pairs. Better yet, these two methods can be applied to translate queries in different situations according to the resources at hand. If we have in-domain parallel corpus, a translation model can be trained to translate the source sentence into target language. And if we only have bilingual dictionaries, the beam-search method also can help to select a set of good translations to form a better query.

4.3 Further Experiments

To further verify the effectiveness of our method in the real comparable data, we further conduct the experiment on the English-Chinese Wikipedia data. We process the English and Chinese Wikipedia data separately and index the Chinese side using the method described in Section 3.1. For each English sentence, the beam-search method is used to translate it into target and here we only keep the best-1 retrieval result. And then we use the coverage matching score with bilingual dictionary to simply extract the sentence pairs higher than a threshold within the retrieval result set. The statistics of sentence pairs over a certain threshold is shown as Table 5. Here H_i means different experiments under different thresholds.

Table 5. Statistics of sentence pairs over a certain threshold

Threshold	**Number of sentence pairs**
H_1: ≥ 0.3	372,387
H_2: ≥ 0.4	250,384
H_3: ≥ 0.45	173,615
H_4: ≥ 0.5	132,043

We use the retained sentence pairs as training data to train the SMT system. GIZA++ and grow-diag-and for word alignment, the Moses toolkit with default settings are used to train the System. NIST MT 2003 and NIST MT 2005 are adopted as development set and test set respectively. The SMT evaluation results using different training data are given in Table 6.

Table 6. SMT evaluation results

Training Data	**BLEU**
H_1	14.32
H_2	16.98
H_3	16.14
H_4	16.56

As Table 6 shows, the BLEU point of using training data H_1 is obviously below than the points of the other three. This is mainly because the threshold of H_1 is 0.3 and it will introduce much noise into the training data. However, the SMT evaluation results can prove the effectiveness of our method from another aspect. Using the CLIR framework, we can build a SMT system from the large comparable corpora.

5 Conclusion and Future work

In this paper, we propose two simple and effective query translation methods for the CLIR based candidate sentence pre-selection framework: one is the beam-search word sequence selection method and the other one is the phrase-based simplified translation model. In beam-search word sequence selection method, only bilingual dictionary and monolingual target corpus is needed and a beam-search algorithm is adopted to select a set of word translations. In phrase-based translation model, a simplified translation model is trained to translate the source sentence into target language. Experimental results show that our method can help obtain candidate sentence pairs of high quality which is quite important to the following parallel sentence and fragments' extraction. What's more, our methods can contribute current SMT for two folds: (1) It can help build a SMT system from nothing but a small size of bilingual dictionary; (2) It can help enhance the current SMT performance with additional mined translation resources.

In the future work, we will try to study on the method of how to mining parallel text including both sentences and fragments from the obtained candidate sentence pairs.

Acknowledgement. The research work has been partially funded by the Natural Science Foundation of China under Grant No. 61333018 and the Hi-Tech Research and Development Program ("863" Program) of China under Grant No. 2012AA011101, and also the High New Technology Research and Development Program of Xinjiang Uyghur Autonomous Region under Grant No. 201312103 as well.

Reference

1. Kenneth W. Church and Patrick Hanks. 1990. Word association norms, mutual information, and lexicography. *Computational Linguistics, 1990, 16(1): 22-29.*
2. Pascale Fung and Percy Cheung. 2004a. Mining very non-parallel corpora: Parallel sentence and lexicon extraction vie bootstrapping and EM. *In EMNLP2004, pages 57-63*.
3. Pascale Fung and Percy Cheung. 2004b. Multi-level bootstrapping for extracting parallel sentences from a quasi-comparable corpus. *In COLING 2004, pages 1051-1057*.
4. Myung-Gil Jang, Sung Hyon Myaeng and Se Young Park. 1999. Using Mutual Information to Resolve Query Translation Ambiguities and Query Term Weighting. *In Proceedings of the 37th annual meeting of the Association for Computational Linguistics on Computational Linguistics, 1999*.
5. Chunyang Liu, Qi Liu, Yang Liu and Maosong Sun. THUTR: A Translation Retrieval System. *Proceedings of the 24th International Conference on Computational Linguistics, pages 321-328*.
6. Mirna Adriani. 2000. Using statistical term similarity for sense disambiguation in cross-language information retrieval. *Information Retrieval, 2(1): 71-82, 2000*.
7. Akira Maeda, Fatiha Sadat, Masatoshi Yoshikawa, and Shunsuke Uemura. 2000. Query term disambiguation for web cross-language information retrieval using a search engine. *Proceedings of the fifth international workshop on information retrieval with Asian languages, ACM, 2000: 25-32.*
8. Dragos Stefan Munteanu and Daniel Marcu. 2005. Improving machine translation performance by exploiting non-parallel corpora. *Computational Linguistics, 31(4):477-504.*
9. Dragos Stefan Munteanu and Daniel Marcu. 2006. Extracting parallel sub-sentential fragments from nonparallel corpora. *In Proceedings of the 21st International Conference on Computational Linguistics and the 44th Annual Meeting of the Association for Computational Linguistics, pages 81-88, Sydney, Australia*.
10. Sadaf Rauf and Holger Schwenk. 2011. Parallel sentence generation from comparable corpora for improved SMT. *Machine Translation, 25(4): 341-375.*
11. Jason R. Smith, Chris Quirk, and Kristina Toutanova. 2010. Extracting parallel sentences from comparable corpora using document level alignment. *In Proceedings of the Human*

Language Technologies/North American Association for Computational Linguistics, pages 403-411.

12. Christoph Tillmann. 2009. A Beam-Search extraction algorithm for comparable data. *In Proceedings of ACL, pages 225-228.*
13. Ferhan Ture and Jimmy Lin. 2012. Why not grab a free lunch? Mining large corpora for parallel sentences to improve translation modeling. *In HLT-NAACL, pages626-630.*
14. Lu Xiang, Yu Zhou and Chengqing Zong. 2013. An Efficient Framework to Extract Parallel Units from Comparable Data. *In Natural Language Processing and Chinese Computing, pages 151-163.*
15. Dan Ştefãnescu, Radu Ion and Sabine Hunsicker. 2012. Hybrid parallel sentence mining from comparable corpora. *Proceedings of the 16th Conference of the European Association for Machine Translation(EAMT 2012),Trento, Italy.*