zNLP: Identifying Parallel Sentences in Chinese-English Comparable Corpora

Zheng Zhang[1,2]
[1]LIMSI, CNRS, Université Paris-Saclay
Orsay, France
zheng.zhang@limsi.fr

Pierre Zweigenbaum[1]
[2] LRI, Univ. Paris-Sud, CNRS,
Université Paris-Saclay
Orsay, France
pz@limsi.fr

Abstract

This paper describes the zNLP system for the BUCC 2017 shared task. Our system identifies parallel sentence pairs in Chinese-English comparable corpora by translating word-by-word Chinese sentences into English, using the search engine Solr to select near-parallel sentences and then by using an SVM classifier to identify true parallel sentences from the previous results. It obtains an F1-score of 45% (resp. 43%) on the test (training) set.

1 Introduction

Parallel sentences are used in many natural language processing applications, particularly for automatic terminology extraction (Lefever et al., 2009) and statistical machine translation (Koehn, 2005; Callison-Burch et al., 2004). However, such resources are scarce for many language pairs and domains. Comparable corpora are sets of texts in two or more languages that are selected according to similar specifications, but are not translations of each other (Sharoff et al., 2013; Morin et al., 2015). Nevertheless, parallel sentences, i.e., sentence pairs that are good translations of each other, can occur naturally in such corpora. Therefore many approaches have been proposed to spot parallel sentences in comparable corpora (Munteanu et al., 2004; Smith et al., 2010).

Extracting parallel sentences from comparable monolingual corpora is a very challenging task. According to the shared task web page,[1] *The aim of the Building and Using Comparable Corpora (BUCC) 2017 shared task is to quantitatively evaluate competing methods for extracting parallel sentences from comparable monolingual corpora, so as to give an overview on the state of the art and to identify the best performing approaches.* More precisely, *given two sentence-split monolingual corpora, the task is to identify pairs of sentences that are translations of each other.*

The BUCC 2017 shared task on parallel sentence extraction raises the following three main issues. One is the cross-language problem: as one must compare sentences across languages (here English with German, French, Russian, or Chinese), one must find a way to compare sentences in two different languages, for instance by first translating one language into the other. Another issue is sentence similarity: how do we define and calculate sentence similarity? The last issue is the existence of too many possible sentence combinations: theoretically, for each sentence in a source monolingual corpus, every sentence in the target monolingual corpus could be used to generate a source-target sentence pair for subsequent parallel sentence identification, which would create a quadratic number of candidate sentence pairs.

Previous work (Smith et al., 2010; Munteanu and Marcu, 2005) on parallel sentence extraction from comparable corpora has used external clues for this purpose. (Smith et al., 2010) bootstrapped the process with document-level sentence alignment. (Munteanu and Marcu, 2005) leveraged the publication date of newspaper articles to trim down the number of candidate sentence pairs. These selection methods are not suitable for the BUCC 2017 shared task as no meta-information is provided on the documents from which the corpus sentences are extracted. In this context, we test how similar methods fare without any meta-information.

In this paper, we describe the system that we developed for the BUCC 2017 shared task and show that a translating-searching-classifying three-step approach can achieve promising results

[1] https://comparable.limsi.fr/bucc2017/bucc2017-task.html

Proceedings of the 10th Workshop on Building and Using Comparable Corpora, pages 51–55,
Vancouver, Canada, August 3, 2017.

for Chinese-English Comparable Corpora.

2 Proposed Method

To address the three problems of the BUCC 2017 shared task, we propose a method which contains three main steps:

1. 'Translating' the monolingual ZH corpus into English.
2. Searching for candidate source-target parallel sentence pairs.
3. Classifying candidate source-target sentence pairs to find parallel sentences.

Note that in our case, the source data is a monolingual English (henceforth EN) corpus and the target data is a monolingual Chinese (henceforth ZH) corpus.

2.1 'Translating' the monolingual ZH corpus into English

To obtain a translated monolingual ZH corpus, a naive approach has been used: we use the Chinese word segmentation tool *jieba* (v0.38)[2] for word segmentation of all the sentences in the monolingual ZH corpus; then we translate these sentences into English word by word with Chinese-English dictionary resources.

The reason for using jieba is that it supports both traditional Chinese and simplified Chinese, which suits our case as the monolingual ZH corpus contains both types of Chinese characters. Besides, jieba has been widely used and could help users obtain good performance in their systems (Shi et al., 2016; Liu et al., 2015; Zhang et al., 2015).

The Chinese-English dictionary resources are *CC-CEDICT*[3], which contains 54,170 traditional Chinese-simplified Chinese-English entries, and the Chinese-English Translation Lexicon Version 3.0 [LDC2002L27] (Huang et al., 2002), which contains 115,128 simplified Chinese-English entries. The merged Chinese-English dictionary contains 196,398 traditional Chinese-English and simplified Chinese-English entries in total. Additionally, for the words not in these two Chinese-English dictionary resources: we keep the original word as its own translation for the words that only contain ASCII characters, and the *Microsoft Translator Text API*[4] has been used to obtain translations of the rest. If a Chinese word receives more than one translation in this process, we keep all of them in the translated sentence.

Note that each sentence in the monolingual ZH corpus has a unique ID. In the translated monolingual ZH corpus, each translated sentence keeps the same ID as its original sentence.

2.2 Searching for candidate source-target (EN-ZH) parallel sentence pairs

Apache Solr[5] (version 6.5.1) is used as our candidate source-target parallel sentence pairs search engine. Solr is an open-source full-text search engine. To rank documents for a user query, Solr computes the score of each matching document based on the model's algorithm and ranks them on their relative score (Shahi, 2015).

Here, we use the tf.idf retrieval function of Solr and index each sentence in the translated monolingual ZH corpus separately. We search each sentence in the monolingual EN corpus and select the top N results for each to generate candidate source-target parallel sentence pairs. Then we cut off results whose score is below a score threshold.

If N is large or the score threshold is low, there will be too many candidate source-target parallel sentence pairs for the next step. We attempted to decrease the number of candidate source-target parallel sentence pairs without sacrificing too much search engine's performance. In this purpose, we evaluated *success* on the training set: the proportion of the question set for which a correct answer can be found within the top N documents retrieved for each question, depending on (N, score threshold). This evaluation aims to find the best N and score threshold parameters for Solr that will return less candidate source-target parallel sentence pairs but still with a high success at N. We set our requirement to a success of 85%.

2.3 Classifying candidate source-target parallel sentence pairs to find parallel sentences among them

After obtaining candidate source-target parallel sentence pairs from the previous step, we use

[2] https://github.com/fxsjy/jieba

[3] https://cc-cedict.org/wiki/ (downloaded on March 16, 2017)

[4] https://azure.microsoft.com/en-us/services/cognitive-services/translator-text-api/

[5] http://lucene.apache.org/solr/

an SVM (Support Vector Machine) classifier[6] to identify parallel sentence pairs among them. We define the following 4 features, which can be extracted from candidate source-target parallel sentence pairs:

- Source-target sentence length ratio
- Solr rank
- Solr score
- Word overlap number

When calculating the source-target sentence length ratio, issues might be caused by cases where one Chinese word has more than one translation. To avoid this, the target sentence length is counted as the number of Chinese words of the original sentence in the monolingual ZH corpus instead of the translated one. The other three features are extracted by using sentences in the translated monolingual ZH corpus and the monolingual EN corpus.

The candidate source-target parallel sentence pairs generated by using the BUCC 2017 shared task training set serve as training data for the SVM model. More precisely, the training data are the candidate source-target parallel sentence pairs generated by taking all the sentences in the training monolingual EN corpus as queries to the search engine in Step 2 (with the selected N and score threshold parameters). The source-target sentence pairs that exist in the training gold standard have been considered as positive examples, the rest are negative examples.

After training the SVM model, we use this classifier to predict parallel sentences from the candidate source-target parallel sentence pairs generated by using the BUCC 2017 shared task test set.

2.4 Evaluation protocol

We perform three evaluations: two independent evaluations on the training set for Step 2 (Searching for candidate source-target parallel sentence pairs) and Step 3 (Finding parallel sentences in candidate source-target sentence pairs) and one evaluation on the training and test sets for the whole system. The first two evaluations aim to find the best parameters and configurations of their own part. The last one is for investigating the effectiveness and performance of the whole system.

[6]We use the SVC implementation of scikit-learn v0.18, http://scikit-learn.org/stable/

For the evaluation of Step 2, we use all the English sentences of the training data gold standard as the question set. According to the success evaluation result, we select the parameter N that provides the required success of 85%. Then a Solr score threshold is calculated as the highest threshold that maintains the success on top N.

To find the best configuration (kernel, class_weight, C, gamma parameters) of the SVM classifier, we perform a 5-fold cross-validation on the training data. As the training data (as well as the test data) is highly imbalanced (the number of negative examples is around 120 times higher than the number of positive examples), the class_weight parameter, according to the scikit-learn web page, which *sets the parameter C of class i to class_weight[i]*C for the SVM classifier*, plays an important role.

For the whole system evaluation, after obtaining the final predicted source-target parallel sentence pairs, we use precision, recall and F1-score as evaluation measures:

$$P = \frac{TP}{TP + FP}; R = \frac{TP}{TP + FN}; F1 = \frac{2PR}{P + R}$$

where TP stands for the number of source-target sentence pairs that is present in the gold standard, a false positive FP is a pair of sentences that is not present in the gold standard and a false negative FN is a pair of sentences present in the gold standard but absent from systems results. We tested three configurations:

1. The standard three-step method.
2. Setting N to 1 and replacing the classifier (Step 3) with a baseline ranking method based on the Solr score: we select the M sentence pairs with the highest scores, where M is determined according to the prior probability of being a correct sentence pair, estimated on the training data.
3. The intersection of Configuration 1 and of Configuration 2, with M=10,000.

3 Results and discussion

The success obtained for the training data is shown in Figure 1. We note that the success is close to 85% when we retrieve the top 3 target sentences ($N = 3$) for each source sentence of the gold standard. If we increase N by 1, 88,860 more negative examples (the number of monolingual EN

sentences in the training corpus) are added to the SVM classifier's training data, but the success improvement is small. We therefore decided not to increase N and set it to 3. Then the maximum Solr score threshold that does not significantly change the success when $N = 3$ is found to be 15.4.

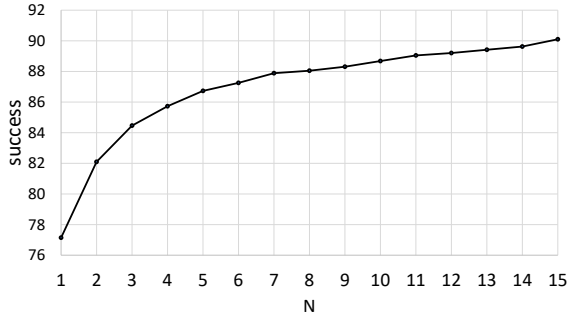

Figure 1: Study of success in the training corpus (evaluation of Step 2)

The results obtained on the training and test sets are presented in Table 1. They are consistent across datasets (test and train): we did not overfit the training set. Our best configuration of SVM classifier for the training data, namely, kernel=rbf, class_weight=1:8, C=1.0, gamma='auto', achieves nearly 0.4037 for precision, 0.4718 for recall and 0.4348 for F1-score. Replacing the classifier with a baseline ranking method only based on the Solr score (Run2) decreases precision, recall and F1-score to 0.2254. This illustrates that only using the tf.idf-based Solr score is not sufficient for the task. Besides, as could be expected, Run 3, the intersection of Runs 1 and 2 is more precise, but incurs a strong decrease in recall. Its recall remains higher than that of Run 2 because it uses a higher M.

Corpus	P	R	F
Training: Run 1	0.4037	**0.4718**	**0.4348**
Training: Run 2	0.2254	0.2254	0.2254
Training: Run 3	**0.4416**	0.4053	0.4227
Test: Run 1	0.4247	**0.4815**	**0.4513**
Test: Run 2	0.2296	0.2300	0.2298
Test: Run 3	**0.4529**	0.4161	0.4338

Table 1: Evaluation results: Run 1 = three steps, Run 2 = no classifier, Run 3 = intersection

We also performed experiments without using the Microsoft Translator Text API. In that case, there is no big change in success. On the test set, with the standard three-step method, this increased recall (0.5153) but decreased precision (0.3158) and F1-score (0.3916).

The whole system does not require external resources other than a Chinese-English dictionary. It is fast: 'translating' the monolingual ZH corpus takes around 1 minute; searching for candidate source-target parallel sentence pairs takes less than 5 minutes for the whole monolingual ZH corpus in the training or test data; the final SVM classifier takes around 20 minutes for training but less than 5 minutes for feature extraction and source-target parallel sentence pairs prediction after obtaining the trained SVM model. However, as the first step's translation is at the word level instead of the sentence level, and for one Chinese word, there are 4.67 English translations on average, we may lose context information of the original words and sentences in the monolingual ZH corpus.

4 Conclusion

In this paper we described the zNLP system for the BUCC 2017 shared task. We proposed a three-step approach to parallel sentence identification in Chinese-English Comparable Corpora by 'translating' the monolingual ZH corpus into English, filtering out candidate parallel sentence pairs with Solr and then selecting the final parallel source-target sentence pairs by using an SVM classifier. Our system identifies parallel sentences with an F1-score of 45.13% in the test data. The proposed method is fast and does not rely on external resources except a Chinese-English dictionary. The code is publicly available at https://github.com/zzcoolj/Parallel-Sentences-Identifier.

Potential pathways for future work include adding more filter conditions to Step 2 (e.g sentence length ratio, word overlap threshold) for candidate source-target parallel sentence pairs. Another pathway would be to add more features to the SVM model. Also in our system, we obtain candidate sentence pairs by searching each sentence in the monolingual EN corpus after indexing each sentence in the translated monolingual ZH corpus separately. We plan to do the reverse (searching sentences in the translated monolingual ZH corpus and indexing the monolingual EN corpus) and combine the two results as our new candidate source-target parallel sentence pairs. We also plan to extend our system to other language pairs by using the relevant dictionaries or word-aligned parallel corpora.

References

Chris Callison-Burch, David Talbot, and Miles Osborne. 2004. Statistical machine translation with word-and sentence-aligned parallel corpora. In *Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics*. Association for Computational Linguistics, page 175.

Shudong Huang, David Graff, and George Doddington. 2002. *Multiple-Translation Chinese Corpus [LDC2002T01]*. Linguistic Data Consortium, Philadelphia. Web download file.

Philipp Koehn. 2005. Europarl: A parallel corpus for statistical machine translation. In *MT summit*. Citeseer, volume 5, pages 79–86.

Els Lefever, Lieve Macken, and Veronique Hoste. 2009. Language-independent bilingual terminology extraction from a multilingual parallel corpus. In *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics*. Association for Computational Linguistics, pages 496–504.

Xuebo Liu, Shuang Ye, Xin Li, Yonghao Luo, and Yanghui Rao. 2015. Zhihurank: A topic-sensitive expert finding algorithm in community question answering websites. In *International Conference on Web-Based Learning*. Springer, pages 165–173.

Emmanuel Morin, Amir Hazem, Florian Boudin, and Elizaveta Loginova Clouet. 2015. LINA: Identifying comparable documents from Wikipedia. In *Eighth Workshop on Building and Using Comparable Corpora*.

Dragos Stefan Munteanu, Alexander M Fraser, and Daniel Marcu. 2004. Improved machine translation performance via parallel sentence extraction from comparable corpora. In *HLT-NAACL*. pages 265–272.

Dragos Stefan Munteanu and Daniel Marcu. 2005. Improving machine translation performance by exploiting non-parallel corpora. *Computational Linguistics* 31(4):477–504.

Dikshant Shahi. 2015. Solr scoring. In *Apache Solr*, Springer, pages 189–207.

Serge Sharoff, Reinhard Rapp, and Pierre Zweigenbaum. 2013. Overviewing important aspects of the last twenty years of research in comparable corpora. In Serge Sharoff, Reinhard Rapp, Pierre Zweigenbaum, and Pascale Fung, editors, *Building and Using Comparable Corpora*, Springer, Berlin Heidelberg, pages 1–20.

Hongjie Shi, Takashi Ushio, Mitsuru Endo, Katsuyoshi Yamagami, and Noriaki Horii. 2016. A multichannel convolutional neural network for cross-language dialog state tracking. In *Spoken Language Technology Workshop (SLT), 2016 IEEE*. IEEE, pages 559–564.

Jason R Smith, Chris Quirk, and Kristina Toutanova. 2010. Extracting parallel sentences from comparable corpora using document level alignment. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*. Association for Computational Linguistics, pages 403–411.

Xiang Zhang, Junbo Zhao, and Yann LeCun. 2015. Character-level convolutional networks for text classification. In *Advances in neural information processing systems*. pages 649–657.