Proceedings of the Ninth International Conference on Machine Learning and Cybernetics, Qingdao, 11-14 July 2010

APPROACHES TO IMPROVING CORPUS QUALITY FOR STATISTICAL MACHINE TRANSLATION

PENG LIU, YU ZHOU, CHENG-QING ZONG

National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China
E-MAIL: {pliu, yzhou, cqzong}@nlpr.ia.ac.cn

Abstract:

The performance of a statistical machine translation (SMT) system heavily depends on the quantity and quality of the bilingual language resource. However, the pervious work mainly focuses on the quantity and tries to collect more bilingual data. In this paper, we aim to optimize the bilingual corpus to improve the performance of the translation system. We propose methods to process the bilingual language data by filtering noise and selecting more informative sentences from the training corpus and the development corpus. The experimental results show that we can obtain a competitive performance using less data compared with using all available data.

Keywords:

Data selection; Noise filter; Corpus optimization; Statistical machine translation

1. Introduction

Statistical machine translation model heavily relies on bilingual corpus which consists of sentences in the source language and their respective reference translation in the target language. In this method, the information of probability is extracted from the training data, and the translation parameters are tuned on the development data. The target language sentence is generated base on the probabilities and parameters. Typically, the more data is used in the training and tuning process, the probabilities and parameters we get will be more accurate and lead to better performance. However, on the one hand massive data will cost more computational resources, and there is much noise in the corpus. On the other hand, in some specific applications such as the translation system running on a mobile or a PDA, the computational resource is limited and a compact and efficient corpus is expected.

For the training data, one problem is how to filter the noise - the wrongly aligned sentence pair. Obviously, the noise will cause wrong word-alignments and reduce the performance of the translation results. The other problem is the scale of the training data. The large amount of training data increases the computational processing load. In the real application, this will reduce the translation speed.

For the development data, the size is much smaller than the training data and the noise could be ignored. The main problem is how to select the most informative sentences to tune the translation parameters. Typically, we run the minimum error rate training (MERT) on the development data [1]. The MERT will search for the optimal parameters by maximizing the BLEU score. But what kind of sentence pairs is suitable for the MERT is still uncertain.

In this paper, we describe approaches to process the training data and the development data, respectively. We filter the noise in the training data using the length ratio and translation ratio methods. And we estimate weight of sentence based on the phrases it contained. The compact training corpus is build according to the sentence weight. For the development data, we select sentences based on surface feature and deep feature, phrase and structure. For both corpora, we verify the relationship between size and translation performance.

The remainder of this paper is organized as follows. Related work is presented in Section 2. The data optimization methods for training corpus and development corpus are described in Section 3 and Section 4. We give the experimental results of these approaches in Section 5 and come to the conclusions in Section 6.

2. Related work

The previous researches on training data and development data mainly focused on the data collection. The researchers tried to get more parallel data for training. Resnik and Smith extracted parallel sentences from web resource [2]; Snover et al. improved the translation performance using comparable corpora [3].

The data selection has been studied by many researchers. Eck et al. selected informative sentences based on n-gram coverage [4]. They used previously unseen n-grams contained in the sentence to measure the importance of the sentence. But they only considered the quantity of the unseen n-grams

978-1-4244-6525-5/10/$26.00 ©2010 IEEE

and didn't take the weight of n-gram into account. Lü et al. selected data for training corpus by information retrieval method [5]. They assumed that the target test data was known before building the translation model, and selected sentences similar to the test text using TF-IDF. The limitation of this method was that the test text must be known first. Yasuda et al. used the perplexity as the measure to select the parallel translation pairs from the out-of-domain corpus; and they integrated the translation model by linear interpolation [6]. Matsoukas et al. proposed a discriminative training method to assign a weight for each sentence in the training set [7]. They limited the negative effects of low quality training data. Liu et al. selected sentences for development set according to the phrase weight estimated from the test set, the method can't be employed if we don't have the test text [8].

As mentioned above, most work focused on the training data, a little of work focused on the development set. However, we pay attention to the data both in training set and development set. The high quality sentence pairs are chosen to construct the translation model and tune the translation parameters.

3. Data processing for training data

In SMT system, the noise in training data seriously affects the quality of the word alignment, imports a lot of errors into the translation rules and reduces the performance of the translation system. However, the translation model, such as the phrase-based translation model heavily depends on the quality of word-alignment. So it is a very basic and important task to filter the noise in the training corpus.

Another problem of the training corpus is the size. Typically, the more data will lead to better performance. But the more data will also cost more computer resource and reduce the translation speed. We have to keep a balance between the performance and the speed.

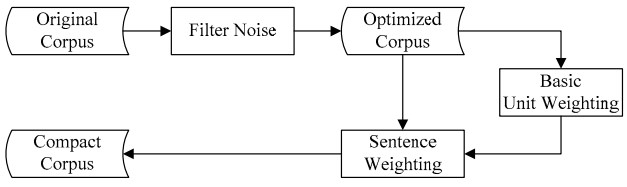

Figure 1. The framework of train data processing

For these two problems, we deal with the training data in two steps. First, we filter the noise in the training data. This new corpus is called as optimized corpus. Second, we estimate the sentence weight by the weight of basic unit, and we select the more informative sentence from the optimized corpus to build a compact training set. We use the compact training set to build a translation system with little performance losing comparing to using all the training data. The framework of the data processing for training data is shown in Figure 1.

3.1. Filter noise methods

In order to filter the noise in the training data, we apply two simple policies: the length ratio (LR) policy and the translation ratio (TR) policy.

LR Policy: Filter the noise by the length ratio. The bilingual sentence pair's length ratio is a simple and effective feature to filter the noise. If a sentence pair is parallel, the length ratio of the two sentences which are corresponding should not differ too much and range in a special bound. So it is a reasonable idea that using the length ratio to filter the noise. Sentence pair whose length ratio is out of the bound will be discarded.

TR Policy: Filter the noise by the translation ratio. We can also use the bilingual dictionary to judge whether the bilingual sentence pair is correctly aligned or not. If two sentences are corresponding, the translation of the word in the source language sentence has a large probability to appear in the target language sentence. We can use a bilingual dictionary to estimate how many word pairs occurred in the two sentences. The translation ratio is defined as following.

$$TR = \frac{\sum \#(word - pair)}{|s|_{src}} \quad (1)$$

where $|s|_{src}$ is the length of the source language sentence; and $\sum \#(word - pair)$ denotes the total number of words whose translations are also appeared in the target language sentence. According to the distribution of the translation ratio from a large scale corpus, we choose thresholds to filter the noise.

We also filter the original training corpus by combing the two policies described above to get a better result. First we filter the noise by the LR of sentence pair, and then we filter the noise according to their TR. The experimental results will be shown in Section 5.

3.2. Data selection method

The methods described above filter the noise in the training corpus. In order to reduce the size of the training data, we also want to select sentences which can cover more information of the entire original corpus.

In information theory, the information contained in a statement is measured by the negative logarithm of the probability of the statement [9][10]. So we can select such information as a feature to estimate the weight of a sentence. Since the phrase-based translation model (PBTM) takes the phrase as the basic translation unit [11], it is a natural idea to

estimate the weight of a sentence according to the information contained in the phrases consisted of such sentence. First, we need to estimate the weight of each phrase, and then estimate the weight of sentence based on those phrases.

As mentioned above, the information contained in a phrase should be calculated by formula (2):

$$I(f) = -\log p(f) \tag{2}$$

where f is a phrase, and $p(f)$ is the probability of the phrase in the corpus. In PBTM, the longer phrase will lead to better performance, so we take the length of phrase into account to construct the weight. And we assign weigh to each phrase using formula (3):

$$w(f) = \sqrt{|f|} \cdot I(f) \tag{3}$$

where $|f|$ is the length of the phrase. We use the square root of the length because of the data smoothing. In order to cover more phrases in the new corpus, we should assign higher weight to the sentence which has more unseen phrases. The weight of each sentence is defined by following formula:

$$w(s) = \frac{\sum_i w(f_i)}{|s|} \tag{4}$$

where s is a sentence, its length is $|s|$, and f_i is the phrase contained in the sentence but not contained in the new corpus, that is to say, the unseen phrase. If a phrase has occurred in the new corpus, the weight is set to zero. If we only consider the new phrases, the longer sentence will tend to get higher score because it contains more unseen phrases. So we divide the score by the sentence length to overcome this problem.

4. Data selection for development data

The development corpus is used to tune the translation parameters which have great influence on the quality and robustness of the translation results. In order to get the optimized parameters, the minimum error rate training is usually employed on the development set. The MERT often consumes too long time and too much computer resources until it converges, especially when the development set is in a large scale. It is a practical requirement to select appropriate size of development set for the MERT. Moreover, it is still a difficult problem what kind and what scale of the development set that used to tune the parameters can achieve an optimal and robust performance. In most cases, one test set translated under parameters tuned by a development set may get a better BLEU score but a worse BLEU score under parameters tuned by another development set.

For the development corpus is often much smaller than the training corpus, we can extract effective features for the data. An intuitive idea is if the extracted sentences can cover more information (such as word, phrase and structure) of the original development set, the new development set will perform better. So we select such sentences which can cover more information of the entire corpus. Because the word is a special phrase, we mainly focus on the phrase- coverage and structure-coverage to introduce our methods.

4.1. Phrase-coverage-based method

As described in training corpus data selection, the phrase is an important feature for PBTM. So we take the phrase coverage as the metric and call this method as phrase-coverage-based method (PCBM).

We take two aspects into account to estimate the weight of phrase: the information it contains and the length of the phrase. The definition of the phrase weight is just as same as the data selection method for training data, see formula (3). One important difference is while estimating the weight of sentence, all the phrases contained in it are considered, not just consider the unseen phrases. In order to avoid the data sparseness problem, we only use the phrase whose length is not longer than four.

The new development data is selected according to the scores of the sentences. Higher score sentence which contains more phrases should be priority selected.

4.2. Structure-coverage-based method

The PCBM only uses phrase, a surface feature, to estimate the weight of sentences. In order to cover more information of the original corpus, we also use some deep features, such as the sentence structure to choose the development set. We name this method as structure-coverage-base method (SCBM).

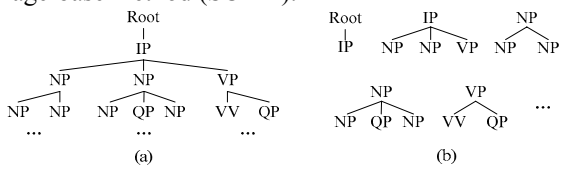

Figure 2. (a) Phrase-structure tree; (b) Subtrees (depth=2).

In this method, we want to extract sentences which can cover the majority structures of the development set. We first parse the entire development corpus into phrase- structure trees. Then we analyze the subtrees contained in the phrase-structure tree, and extract sentences which can cover more subtrees. In order to avoid the data sparseness problem, we use the subtree whose depth is between two and four. An example of the subtree is shown in Figure 2.

We consider two aspects to estimate the weight of the subtrees: depth and information. For a subtree t, and its depth is $|t|$, let's assume its probability is $p(t)$, which is estimated from the development set, and the information

contained in it is calculated by formula (5).

$$I(t) = -\log p(t) \tag{5}$$

Then the weight of each subtree in the development set is calculated by formula (6):

$$w(t) = \sqrt{|t|} \cdot I(t) \tag{6}$$

And the score of sentence s is calculated by the following formula:

$$S_{SCBM}(s) = \frac{\sum_{t \in T} w(t)}{|s|} \tag{7}$$

where T is the set of subtrees contained in the sentence s, and $S_{SCBM}(s)$ is the score of the sentence using the SCBM. Then we can select the sentences according to their scores, the higher score sentence has more information.

5. Experimental results

In our experiments, we use MOSES[1] as our translation engine. The translation results are evaluated by the BLEU metrics [12].

5.1. Results on training corpus selection

On training data processing, we did our experiments on the CWMT 2008 (China Workshop on Machine Translation)[2] corpus. We randomly choose 20 million words as the original training corpus to construct our experiments on Chinese-to-English translation task. And we randomly select 400 sentences from the development set as the test set.

- Experiments on filtering noise method

In our experiments, we find that the length ratio of more than 96% sentence pairs are between 0.6 and 1.7, so we take these two values as thresholds. The sentence pair whose length ratio is out of this bound will be discarded.

Then we filter the noise based on the TR policy. First, we obtain the lemma of each word in target language by using morph toolkit[3]. We use a dictionary which contains more than 950 thousands words to calculate the translation ratio of each sentence. In the training corpus, the translation ratio of about 98% sentence pairs is higher than 0.2. So we take this value as the threshold. The sentence pair whose translation ratio is less than 0.2 is regarded as noise data. Then we build the translation model on the new training corpora. The experimental results are shown in Table 1.

TABLE 1. RESULTS OF NOISE FILTER FOR THE TRAINING CORPUS

	All	**LR**	**TR**	**Comb**
Words(M)	20.00	19.67	19.72	19.40
BLEU	0.2132	0.2128	**0.2153**	0.2135

The second column is the result of using all original training data. The LR is the result of the corpus filtered by the LR method. About 0.33 million words are filtered and the BLEU score has 0.04% decline. The forth column is the result of the corpus filtered by the TR method. About 0.28 million words are filtered and the BLEU score improves 0.21%. The last column is the result of corpus filtered by combing these two methods. The BLEU score is almost as the same as the original corpus with 0.6 million words filtered. From the Table 1, it is clear that the TR method is more robust and effective. This is because of that the method makes use of the bilingual dictionary information, and the precision is higher than the result of the LR method. The LR method can get a competitive performance compare to using all the training data.

- Experiments on training data selection methods

In the data selection experiments, we select different size of corpus to combine the new training corpus. We tried three methods: 1) We select sentences randomly from the training data to combine the baseline system; 2) We weigh the sentences only considering the quantity of the unseen phrases without considering the weight of phrases. This method is called *unWP*. 3) We consider both the quantity and weight of the unseen phrase, this method is called *WP*.

The BLEU score, recall of words and percentage of sentences of the experimental results are presented in Table 2. From the results, it is clear that the data selected using our method could cover more phrases and get a higher score using small size data. We could get the competitive performance with much less data, and this will reduce the computational load. For example, when we use only half of the training data, the baseline only covers 45.9% words, while the unWP method could cover 91.8% words and the WP method could cover 92.3% words. The word coverage is much higher than the baseline. The BLEU score of WP method is 0.2060, 5.28% higher than the baseline 0.1532, and it is only 0.72% lower than the system that using all the available data with its score 0.2132. And the corresponding training corpora have almost the same quantity of sentences. The sentence percentages are 46.5%, 45.2% and 45.8%, respectively. We use only half of the data to get a competitive performance compared to using all the data.

[1] http://www.statmt.org/moses/

[2] http://nlpr-web.ia.ac.cn/cwmt-2008/

[3] http://www.informatics.susx.ac.uk/research/groups/nlp/carroll/morph.html

TABLE 2. RESULT OF DATA SELECTION FOR TRAINING DATA

Words(M)	Baseline			unWP			WP		
	BLEU	Recall	Percent	BLEU	Recall	Percent	BLEU	Recall	Percent
2	0.1357	24.8%	9.3%	0.1614	55.7%	7.5%	0.1726	67.0%	8.2%
4	0.1384	30.6%	18.7%	0.1842	74.9%	16.5%	0.1918	78.7%	17.2%
6	0.1468	36.8%	27.9%	0.1887	83.1%	25.7%	0.1955	85.1%	26.5%
8	0.1511	42.6%	37.1%	0.1947	88.3%	35.3%	0.2010	89.2%	36.0%
10	0.1532	45.9%	46.5%	0.2033	91.8%	45.2%	0.2060	92.3%	45.8%
12	0.1609	51.1%	55.7%	0.2059	94.4%	55.2%	0.2071	94.6%	55.8%
14	0.1724	60.4%	65.3%	0.2055	96.3%	65.5%	0.2098	96.4%	66.0%
16	0.1990	82.8%	77.7%	0.2100	97.9%	76.3%	0.2118	97.9%	76.7%
18	0.2095	95.7%	89.1%	0.2046	99.2%	87.3%	0.2121	99.2%	87.6%
20	0.2132	100.0%	100.0%	---	---	---	---	---	---

With the same size of data, our method could extract more informative sentences and cover more words. The training data selected using unWP and WP methods both perform better than the baseline system, especially when the training data in small size. Comparing to the unWP, we can reach a higher performance when we take the weight of phrase into account.

5.2. Results on development corpus selection

We did the data selection experiments for development corpus on CWMT 2009[4] and IWSLT 2009[5] translation tasks, both in bidirectional translation for Chinese and English. The former is in news domain and the latter is in travel domain. For CWMT 2009 task, we randomly select 400 sentences from the development set as the test set, and take the left as the development set. For IWSLT 2009 tasks, we employ BTEC Chinese-to-English task and Challenge English-to-Chinese task. The Table 3 shows the information of the corpora.

TABLE 3. DEVELOPMENT DATA FOR DATA SELECTION

	Task	Development set		Test set
		Sen	Words	Sen
CWMT 2009	C-E	2,876	57,010	400
	E-C	3,081	55,815	400
IWSLT 2009	C-E	2,508	17,940	469
	E-C	1,465	12,210	393

On each task, we select sentences randomly to build the baseline. Then we selected the different scale of development data for the MERT using the approaches we proposed. For the PCBM method, we consider the phrases from the Chinese sentences (Ch), the English sentences (En) and both of them (Ch+En). For the SCBM method, we only use the Chinese sentences and parse them using the Stanford parser [13]. The results are shown in Figure 3.

In these figures, the horizontal axis is the scale of the development corpus, the unit is thousand words. The vertical axis is the BLEU score of the test set using the parameters trained on the corresponding development data. From these experiments, it is clear that comparing to the baseline system, the development corpus selected using our methods can get higher performance with the same quantity of data. When the development corpus is in large scale, our method can select more informative sentences for MERT. For the phrase-coverage-based method, when consider both the Chinese phrase and the English phrase, the performance is better and more robust comparing to the methods which only consider monolingual phrase. This is because the sentences extracted using this method could cover the information both in source language and target language, and make the translation parameters more robust.

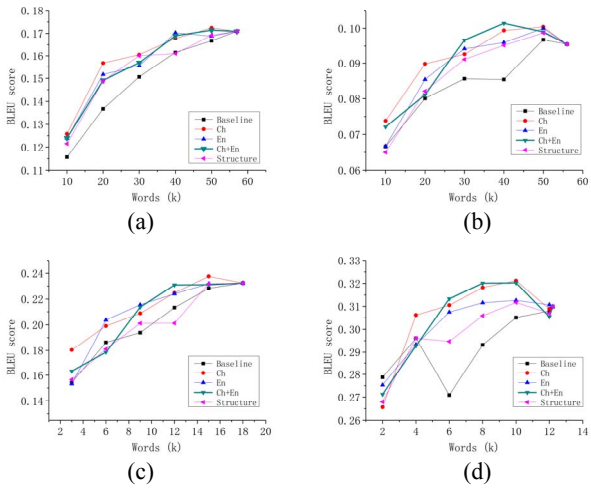

Figure 3: Results of data selection for development data:
(a) CWMT09 C-E; (b) CWMT09 E-C;
(c) IWSLT09 C-E; (d) IWSLT09 E-C.

The structure-coverage-based method performs not as good as the phrase-coverage-based method, though it is better

[4] http://www.icip.org.cn/cwmt2009

[5] http://mastarpj.nict.go.jp/IWSLT2009/

than the baseline. This is because that the precision of the parser is not good enough. The parser will import many errors into the parsing results and decrease the performance of the translation system. For this reason, we didn't try combination of phrase-coverage-based method and the structure-coverage-based method. The former method has a higher and more robust performance.

Another notable phenomenon is that we can get even higher score using a part of the development data than using all the data. For example, in Figure 3-d), when we using 10 thousand words for MERT, the performance is better than using 12 thousands words. We present the recall of words for the baseline method and the PCBM method which considers bilingual phrases in Table 4. From this table, the baseline's recall is only 77.0% while the PCBM's recall is 99.9% when the development data has 10 thousand words; almost all the words have been covered. Adding more data to the development set brings little improvement to the recall of words, but imports much redundancy sentences and reduces the performance of the translations.

TABLE 4. RECALL OF WORDS FOR IWSLT09 E-C

Words (k)	2	4	6	8	10	12	12.21
baseline	0.266	0.413	0.508	0.559	0.770	0.978	1.000
Ch+En	0.552	0.802	0.932	0.984	0.999	1.000	1.000

6. Conclusions

The performance of the SMT system heavily depends on the quality and quantity of the corpus. In this paper, we propose approaches to improve the quality of the training corpus and the development corpus. For the training corpus, we filter the noise based on the length ratio and the translation ratio policies. Then we select more informative sentences to build a compact training corpus using the weighted-phrase method. For the new compact training corpus, we can get a competitive performance compared to the baseline system using all training data.

The data selection for development corpus using two kinds of features: the phrase and the structure. The experimental results show that both methods perform better than the baseline. When consider the bilingual phrases, the performance is better and more robust. The PCBM is better than the SCBM. One reason is that the parser could import errors to the parsing tree and there exists serious data sparseness problem in syntax structures; the other reason is the translation engine is phrase-based translation system, it could not make full use of the information contained in the phrase-structure tree.

Acknowledgements

The research work has been partially funded by the Natural Science Foundation of China under Grant No. 6097 5053, 90820303, and 60736014, the National Key Technology R&D Program under Grant No. 2006BAH03B02, the Hi-Tech Research and Development Program ("863" Program) of China under Grant No. 2006AA010108-4, and also supported by the China-Singapore Institute of Digital Media (CSIDM) project under grant No. CSIDM-200804.

References

[1] Och F. J., "Minimum Error Rate Training in Statistical Machine Translation", Proc. of the 41st ACL, Sapporo, pp. 160-167, Jul. 2003.

[2] Resnik P., and N. A. Smith, "Articles The Web as a Parallel Corpus", Computational Linguistics, Vol 29, No. 3, pp. 349-380, 2003.

[3] Snover M., B. Dorr, and R. Schwartz, "Language and Translation Model Adaptation using Comparable Corpora", Proc. of EMNLP, pp. 857-866, Oct. 2008.

[4] Eck M., S. Vogel, and A. Waibel, "Low Cost Portability for Statistical Machine Translation based on N-gram Coverage", Proc. of the 10th MT Summit, Phuket, Thailand, pp. 227-234, Sep. 2005.

[5] Lü Y., J. Huang, and Q. Liu, "Improving Statistical Machine Translation Performance by Training Data Selection and Optimization", Proc. of EMNLP-CoNLL, Prague, pp. 343-350, Jun. 2007.

[6] Yasuda K., R. Zhang, H. Yamamoto, and E. Sumita, "Method of Selecting Training Data to Build a Compact and Efficient Translation Model", Proc. of the 3rd IJCNLP, India, pp. 655-660, Jan. 2008.

[7] Matsoukas S., A.-V. I. Rosti, and B. Zhang, "Discriminative Corpus Weight Estimation for Machine Translation", Proc. of EMNLP, Singapore, pp. 708-717, Aug. 2009.

[8] Liu P., Y. Zhou, and C. Zong, "Approach to Selecting Best Development Set for Phrase-based Statistical Machine Translation", Proc. of the 23rd PACLIC, Hongkong, pp. 325-334, Dec. 2009.

[9] Cover T. M., and J. A. Thomas, Elements of Information Theory, Wiley, New York, 1991.

[10] Lin D., "An Information-Theoretic Definition of Similarity", Proc. of the 5th ICML, pp. 296-304, 1998.

[11] Koehn P., F. J. Och, and D. Marcu, "Statistical Phrase-Based Translation", Proc. of HLT-NAACL, Edmonton, pp. 48-54, 2003.

[12] Papineni K., S. Roukos, T. Ward, and W.-J. Zhu, "Bleu: a Method for Automatic Evaluation of Machine Translation", Proc. of 40th ACL, pp. 311-318, 2002.

[13] Levy R., and C. D. Manning, "Is it Harder to Parse Chinese, or the Chinese Treebank?", Proc. of the 41st ACL, Sapporo, Japan, pp. 439-446, Jul. 2003.