Self-Training for Mining Parenthetical Translations in Monolingual Web Pages

Xianchao Wu[†] **Naoaki Okazaki**[†] **Jun'ichi Tsujii**[†‡]

[†]Computer Science, Graduate School of Information Science and Technology, University of Tokyo
7-3-1 Hongo, Bunkyo-ku, Tokyo 113-8656, Japan
[‡]School of Computer Science, University of Manchester
National Centre for Text Mining (NaCTeM)
Manchester Interdisciplinary Biocentre, 131 Princess Street, Manchester M1 7DN, UK

{wxc, okazaki, tsujii}@is.s.u-tokyo.ac.jp

1 Introduction

Bilingual lexicons are widely used in applications of multi-lingual language processing, such as machine translation and cross-lingual information retrieval. However, it is hard to maintain a comprehensible bilingual lexicon with the continuous emergence of neologisms (e.g., new technical terms, personal names, abbreviations). Thus, numerous researchers extracted bilingual lexicons from large-scaled corpora such as the Web. In general, two kinds of corpora were used for lexicon mining. One is pure monolingual corpora, where frequency-based expectation-maximization algorithms and cognate clues play a central role (Koehn and Knight, 2002; Haghighi et al., 2008). Another is bilingual parallel and comparable corpora, where co-occurrences and context clues are used (Cao et al., 2007; Lin et al., 2008).

In this paper, we focus on a special type of linguistic expressions which are found in monolingual corpora, *parenthetical translations*. This is motivated by the observation that news and technical papers written in some languages (e.g., Chinese, Japanese) often annotate named entities or technical terms with their translations in English inside parentheses.

A parenthetical translation can be expressed by the following pattern,

$$f_1\ f_2\ ...\ f_n\ (e_1\ e_2\ ...\ e_m). \tag{1}$$

Here, $f_1\ f_2\ ...\ f_n$, the pre-parenthesis text, denotes a word sequence in a target language (other than English); and $e_1\ e_2\ ...\ e_m$, the in-parenthesis text, denotes a word sequence of English that is the translation of $f_1\ f_2\ ...\ f_n$. Parenthetical translations can be separated into several categories: *bilingual abbreviation*, *transliteration*, *translation*, and their mixture. Table 1 illustrates several examples of these categories. In this paper, we only concentrate on the translation category for simplicity.

Type	**Examples with translations in** *italic*
Abbreviation	对 ‖ 全球 气候 观测 系统 (GCOS) *to Global Climate Observing System (GCOS)*
Transliteration	品牌 将 在 ‖ 辛普顿 - 特尔曼 (Shipton-Tilman) *brand will be among Shipton-Tilman (Shipton-Tilman)*
Translation	定时炸弹，‖ 删除蝇 (Cancelbots) *time bomb, Cancelbots (Cancelbots)*
Mixture	在 香港 上课 的 英国 ‖ 布拉福特大学 (Bradford University) *the English Bradford University (Bradford University) that holds lessons in Hongkong*

Table 1: Parenthetical translation categories and examples selected from the Chinese Web pages. Mixture stands for the mixture of translation (*University*) and transliteration (*Bradford*). '‖' denotes the left boundary of each pre-parenthesis text.

Parenthetical translation mining faces the following problems. First, we need to distinguish parenthetical translations from general parenthetical expressions, because parenthesis has various functions (e.g., defining abbreviations, elaborations, ellipsis, citations, annotations) as well as translation. Second, we need to determine the left boundary (denoted as ‖ in Table 1) of the pre-parenthesis text.

Heuristic rules have been used for the first problem. For example, Lin et al. (2008) addressed several rules such as the assumption that the pre-parenthesis text is predominantly in Chinese and the in-parenthesis text is predominantly in English. In order to deal with the second problem, supervised (Cao et al., 2007) and unsupervised (Lin et al., 2008) methods have been proposed. Cao et al. (2007) split the mining task into two parts, transliteration detection and translation detection. They used grapheme-based transliteration probabilities as a threshold to extract transliteration entries. Then, they manually annotated the left boundaries of a number of parenthetical expressions to train a classifier for classifying boundary-unknown candidates. Lin et al. (2008) applied a co-occurrence based word alignment approach, Competitive Link[1] (Melamed, 2000), to determine the outer boundary. However, supervised approaches are restricted by the manually annotated training data since it is a time-consuming task for us to annotate the left boundaries for training. And unsupervised approaches are weak dealing with low frequency cases keeping semantic clues unused. For example, the frequency-based approach is helpless for us to determine the boundaries of the examples listed in Table 1 if they appear only once in the available monolingual corpora. Furthermore, unsupervised approaches are based on the assumption that the translation relation holds between a Chinese phrase and an English phrase only if they have a relatively high frequency of co-occurrence.

The challenge is that how can we automatically exact a higher precision and recall lexicon based on semantic information. Dealing with this challenge, we resort to a seed lexicon and propose a semi-supervised learning algorithm. Being able to learn from labeled data and unlabeled data, semi-supervised approaches have been used in numerous applications, such as self-training in word sense disambiguation (Yarowsky, 1995), parsing (McClosky et al., 2008), etc. In this paper, we propose a semi-supervised framework for mining parenthetical translations. The main idea is to make use of a seed lexicon to train a translation model and determine the boundaries employing semantic clues. We employ a cascaded translation model (Wu et al., 2008) by self-training it based on morpheme-level, lexical level, and phrasal level translation probabilities.

2 System Framework and Self-Training Algorithm

Our system framework for mining parenthetical translations includes the following steps. First, parenthetical expressions matching Pattern 1 in Chinese Web pages and documents are extracted. Then, pre-parenthetical Chinese sequences are segmented into words by using a Chinese word segmentor, S-MSRSeg[2] (Gao et al., 2006). Third, the initial parenthetical translation corpus is constructed by applying the heuristic filtering rules defined in (Lin et al., 2008). In addition, we manually defined a filtering list which included about 50 words (mainly copulas, verbs, and prepositions) and punctuation such as $\ll$, 「, ，, 是, 对, 在. We match the pre-parenthetical text with the filtering list. Expressions appearing before one in the filtering list are dropped (the boundaries of the first three examples in Table 1 can be successfully determined through this way). This pre-processed parenthetical translation corpus is taken as our initial candidate set for mining, hereafter.

Algorithm 1 self-training algorithm

```
Require: $L, U = \{\boldsymbol{u}|\boldsymbol{u} = \boldsymbol{u}_C(\boldsymbol{u}_E)\}, T, M$   ▷$L$, (labeled)
    training set; $U$, (unlabeled) candidate set; $T$, test set; $M$,
    the translation model.

 1: $Lexicon = \{\}$   ▷ new mined lexicon
 2: repeat
 3:     $N = \{\}$ ▷ new mined lexicon during one iteration
 4:     train $M$ on $L$
 5:     evaluate $M$ on $T$
 6:     for $\boldsymbol{u} = \boldsymbol{u}_C(\boldsymbol{u}_E) \in U$ do
 7:         $topN = \{C'|\text{decode } \boldsymbol{u}_E \text{ by } M\}$
 8:         $N = N \cup \{(c, \boldsymbol{u}_E)|c \in \boldsymbol{u}_C \wedge$
                     $\exists C' \in topN \; s.t. \; p(C', \boldsymbol{u}_E) \geq \theta_1 \wedge$
                     $\text{WER}(c, C') \leq \theta_2$
 9:     end for
10:     $U = U - N$
11:     $L = L \cup N$
12:     $Lexicon = Lexicon \cup N$
13: until condition
14: return $Lexicon$   ▷ the output
```

Algorithm 1 presents the self-training algorithm for lexicon mining. The main part is a loop from Line 2 to Line 13. We take a given seed lexicon as labeled data, and split it into training and testing sets (L and T). U=$\{\boldsymbol{u}_C(\boldsymbol{u}_E)\}$ stands for the (unlabeled) parenthetical candidates. Initially, a translation model (M) is trained on L and evaluated on T (Line 4 and 5). Then, the English phrase $\boldsymbol{u}_E$ of each unlabeled entry $\boldsymbol{u} \in U$ is decoded by M, and the top-N outputs are stored in $topN$ (Line 7). In

[1] φ^2 (Gale and Church, 1991) value was used as the link score

[2] http://research.microsoft.com/research/downloads/details/7a2bb7ee-35e6-40d7-a3f1-0b743a56b424/details.aspx

Line 8, c is a substring of $\boldsymbol{u}_C$ and it is generated by deleting zero or more words from the left side of $\boldsymbol{u}_C$. A similarity function on c and a translation output $C' \in topN$ is employed to make the decision of classification: the pair $(c, \boldsymbol{u}_E)$ will be selected as a new entry if the translation probability $p(C', \boldsymbol{u}_E)$ is no less than a threshold θ_1 and the word-error-rate (WER) between c and C' is no larger than a threshold θ_2 (Line 8). After processing each entries in U, the new mined lexicon N is deleted from U and inserted to the current training set L as a new training set (Line 10 and 11). Also, N is included to the final lexicon (Line 12). When some condition is satisfied, the loop stops. Finally, the algorithm returns the mined lexicon in total.

The main idea of Algorithm 1 is that the left boundaries are determined by semantic similarities provided by a self-trained translation model. Indeed, the semantic similarity takes the form of translation probability, which is estimated according to co-occurrence information. We use the morpheme-level translation similarity in a cascaded translation model (Wu et al., 2008), which makes use of morpheme, word, and phrase level translation units. We segment English words into morphemes (sequences of prefixes, stems, and suffixes) by Morfessor 0.9.2[3], an unsupervised language-independent morphological analyzer (Creutz and Lagus, 2007). To gain a morpheme-level translation table, we run GIZA++ (Och and Ney, 2003) on both directions with configuration $1^5H^53^54^5$ between English morphemes and Chinese characters, and take the intersection of *Viterbi* alignments.

We show an example to explain how Algorithm 1 works. Suppose we have a candidate '利用 核苷 二磷酸 糖 (nucleoside diphosphate sugars)' and 'nucleoside diphosphate sugars' is translated into '核苷 二 phosphate 糖' with a translation probability of p_1. 'diphosphate' is a unseen word to our translation model and only the prefix 'di' is translated into '二'. Among the substrings of '利用 核苷 二磷酸 糖', '核苷 二磷酸 糖' is optimal since WER('核苷 二磷酸 糖','核苷 二 phosphate 糖')=0.33 is the smallest. If $p_1 \geq \theta_1$ and $0.33 \leq \theta_2$, then the pair (核苷 二磷酸 糖, nucleoside diphosphate sugars) will be selected as a new lexicon entry.

%	Initial	1	2	3	4	5
100	.1810	.1814	.1973	.1965	**.1990**	.1968

Table 2: The BLEU score of self-trained cascaded translation model under n-th (n=1..5) iteration.

	Precision	Recall
Unsupervised	74.6%	20.7%
Ours	77.7%	28.8%

Table 3: The comparison of our approach and an unsupervised approach (Lin et al., 2008).

3 Experiment

We used SogouT Internet Corpus Version 2.0[4] which contains about 13 billion Chinese Web pages (252 GB text files), and Peking University Chinese Paper Corpus (55 GB text files). We constructed a partially parallel corpus with 12,444,264 entries from the two corpora through filtering by Pattern 1, heuristic rules defined in (Lin et al., 2008), and our manually defined filtering list.

3.1 Self-Training Evaluation

We used Wanfang Chinese-English technical term dictionary[5], which contains 525,259 entries in total, for training and testing. 10,000 entries were randomly selected as a test set and the remaining entries for training. We set the terminal condition in Algorithm 1 to be running constant times; θ_1 to be 1E-20; and θ_2 to be 0.5. Only the top-1 output (C')was used for comparing.

Table 2 illustrates the BLEU scores (Papineni et al., 2002) of the translation model before and after n-th (n=1..5) self-training. After running five times of Algorithm 1, we gained 9.9% relative improvement of BLEU score from 0.1810 to 0.1990. We finally mined 2,916,085 distinct entries.

3.2 Comparison with Unsupervised Approach

We compare our self-training based approach with an unsupervised method (Lin et al., 2008). 2,628,366 distinct entries were obtained after we applied the Competitive Link word alignment algorithm on the same partially parallel corpus. The

[3] http://www.cis.hut.fi/projects/morpho/

[4] http://www.sogou.com/labs/dl/t.html

[5] http://www.wanfangdata.com.cn/Search/ResourceBrowse .aspx

threshold value of the link score (φ^2 value) in Competitive Link was empirically set to be 0.001.

We randomly selected 500 entries from the partially parallel corpus, and then check how many entries were extracted with correct boundaries. Table 3 shows the comparison. Of the 500 entries, 330 are translations, 135 are transliterations, 17 are abbreviations, and only 18 are wrong candidates (i.e., there is no translation relation between the pre-parenthetical text and the in-parenthetical text). This provides a strong evidence that large scale bilingual lexicons do can be mined from parenthetical expressions. We mined 179 entries in which 139 were correct, with a precision of 77.7% and recall of 28.8% which are better than that of the unsupervised approach. This is because we made use of an external lexicon that provided more semantic-level clues.

However, it should be notified that the recalls of our approach is still very small. This is because the low coverage and bias entries of the seed lexicon used. A large amount of English words are not covered by the lexicon, such as transliterations of personal names and places. Only 8.65% English words in the partially parallel corpus are covered by the Wanfang dictionary. On the other hand, 55.2% English words in Wanfang dictionary are not covered by the partially parallel corpus. The result also uncover the drawback of our approach in terms of the strong dependency to a seed lexicon. The future work would be to use a two-stage strategy: first take the high co-occurrence candidates as the seed lexicon, and then self-train a translation model start from the seed lexicon for determining the boundaries of relatively low-frequency candidates.

4 Conclusion

We have proposed a semi-supervised learning framework for mining an English-Chinese lexicon from parenthetical expressions in the Chinese Web pages. The mined lexicon contains 2,916,085 entries with a precision of 77.7%. A self-trained cascaded translation model was used for determining the left-boundaries of the pre-parenthetical texts. Through sample testing, we gained better precision and recalls comparing our semi-supervised framework with our implementation of (Lin et al., 2008)'s unsupervised mining approach.

Acknowledgments

This work was partially supported by Grant-in-Aid for Specially Promoted Research (MEXT, Japan) and Japanese/Chinese Machine Translation Project in Special Coordination Funds for Promoting Science and Technology (MEXT, Japan).

References

Cao, Guihong, Jianfeng Gao, and Jian-Yun Nie. 2007. A system to Mine Large-Scale Bilingual Dictionaries from Monolingual Web Pages. In *MT Summit XI*.

Creutz, Mathias and Krista Lagus. 2007. Unsupervised Models for Morpheme Segmentation and Morphology Learning. *ACM Transactions on Speech and Language Processing*, 4(1):Article 3.

Gale, W. and K. Church. 1991. Identifying word correspondence in parallel text. In *Proceedings of the DARPA NLP Workshop*.

Gao, Jianfeng, Mu Li, Andi Wu, and Chang-Ning Huang. 2006. Chinese Word Segmentation and Named Entity Recognition: A Pragmatic Approach. *Computational Linguistics*, 31(4):531–574.

Haghighi, Aria, Percy Liang, Taylor Berg-Kirkpatrick, and Dan Klein 2008. Learning Bilingual Lexicons from Monolingual Corpora. In *ACL-08:HLT*.

Koehn, Philipp and Kevin Knight. 2002. Learning a translation lexicon from monolingual corpora. In *SIGLEX 2002*.

Lin, Dekang, Shaojun Zhao, Benjamin Van Durme, and Marius Paşca. 2008. Mining Parenthetical Translations from the Web by Word Alignment. In *ACL-08:HLT*.

McClosky, David, Eugene Charniak, and Mark Johnson 2008. When is Self-Training Effective for Parsing? In *COLING 2008*.

Melamed, I. Dan. 2000. Models of Translational Equivalence among Words. *Computational Linguistics*, 26(2):221–249.

Och, Franz Josef and Hermann Ney. 2003. A Systematic Comparison of Various Statistical Alignment Models. *Computational Linguistics*, 29(1):19–51.

Papineni, Kishore, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a Method for Automatic Evaluation of Machine Translation. In *ACL 2002*.

Wu, Xianchao, Naoaki Okazaki, Takashi Tsunakawa, and Jun'ichi Tsujii. 2008. Improving English-to-Chinese Translation for Technical Terms Using Morphological Information. In *AMTA 2008*.

Yarowsky, David. 1995. Unsupervised Word Sense Disambiguation Rivaling Supervised Methods. In *ACL 1995*.