Learning from Parenthetical Sentences for Term Translation in Machine Translation

Guoping Huang and **Jiajun Zhang** and **Yu Zhou** and **Chengqing Zong**
National Laboratory of Pattern Recognition,
Institute of Automation, Chinese Academy of Sciences, Beijing, China
{guoping.huang, jjzhang, yzhou, cqzong}@nlpr.ia.ac.cn

Abstract

Terms extensively exist in specific domains, and term translation plays a critical role in domain-specific machine translation (MT) tasks. However, it's a challenging task to translate them correctly for the huge number of pre-existing terms and the endless new terms. To achieve better term translation quality, it is necessary to inject external term knowledge into the underlying MT system. Fortunately, there are plenty of term translation knowledge in parenthetical sentences on the Internet. In this paper, we propose a simple, straightforward and effective framework to improve term translation by learning from parenthetical sentences. This framework includes: (1) a focused web crawler; (2) a parenthetical sentence filter, acquiring parenthetical sentences including bilingual term pairs; (3) a term translation knowledge extractor, extracting bilingual term translation candidates; (4) a probability learner, generating the term translation table for MT decoders. The extensive experiments demonstrate that our proposed framework significantly improves the translation quality of terms and sentences.

1 Introduction

Terms, the linguistic representation of concepts, a noun or compound word used in a specific context, deliver essential context and meaning in human languages, such as "interprocess communication" or abbreviated as "IPC". In this paper, we do not consider named entities (e.g., person names, location names, organization names, time and numbers). Terms extensively exist in specific domains. For example, in Microsoft Translation Memory, there are 8 terms out of every 100 words, whereas names entities are nearly nonexistent. What's more, new terms are being created all the time, such as in areas of computer science and medicine. Thus, term translation plays a critical role in domain-specific tasks of machine translations (MT), especially statistical machine translation (SMT).

However, unlike person names or other named entities having obvious characteristics and boundary clues, it's a challenging task to translate terms correctly for the huge number of pre-existing terms and the endless new terms. In practice, to achieve better term translation quality, it is necessary to inject external term knowledge into the underlying MT system. The best way is to import a bilingual technical term dictionary, such as such as Microsoft Terminology [1]. But the high cost makes it impossible to construct such bilingual dictionary by human experts for various domains. Thus how to learning bilingual term knowledge automatically becomes the key of term translation in domain-specific MT tasks.

The state-of-art term translation knowledge extraction methods tend to take the Internet as a big corpus (Ren et al., 2010). The most important assumption behind these methods is that the corresponding translation for every source term must exist somewhere on the web. Then, the term translation pair extraction problem is converted to the task of finding these translations from the web and extract them correctly. As a result, except terms, the other various fragments, including multi-word expressions , will be extracted for the lack of term recognization. Not surprisingly, it has increased system workloads and directly reduces the quality of term translation.

[1] https://www.microsoft.com/Language/en-us/default.aspx

Proceedings of the 9th SIGHAN Workshop on Chinese Language Processing, pages 37–45,
Taipei, Taiwan, December 1, 2017.

Example 1: A parenthetical sentence

不过 各个 进程 有 自己 的 内存 空间 、 数据 栈 等 ， 所以 只能 使用 **进程 间 通讯** (interprocess communication , IPC) ， 而 不能 直接 共享 信息 。

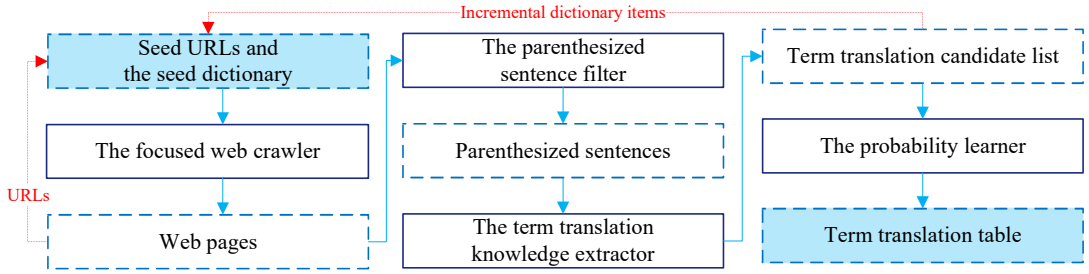

Figure 1: An overview of learning from parenthetical sentences for term translation.

For the extraction of term translation knowledge, we should put quality before quantity. Thus, in this paper, we turn to parenthetical sentences in mix-language web pages for acquiring term translation knowledge. In this work, a sentence will be called **parenthetical sentence** when the following conditions are true: (1) the sentence contains one or more parentheses; (2) the phrase immediately to the left of the parenthesis is a term; (3) the corresponding translation of the term is included in the parenthesis. The parenthetical sentence can be denoted as $s = c_1c_2 \ldots c_n(e_1e_2 \ldots e_m)$, where $c_1c_2 \ldots c_n$ is a Chinese term and $e_1e_2 \ldots e_m$ is its corresponding English translation. In this paper, the term included in the parenthesis and out of parentheses are referred to *source term* ($e_1e_2 \ldots e_m$) and *target term* ($c_1c_2 \ldots c_n$), respectively. A typical parenthetical sentence is shown as following Example 1.

In Example 1, the Chinese sentence contains one parenthesis, the phrase "进程 间 通讯" immediately to the left of the parenthesis is a target term, and the corresponding source term is "interprocess communication" or abbreviated as "IPC". Therefore, it is a parenthetical sentence.

There are plenty of term translation knowledge in parenthetical sentences. Compared with parallel/comparable sentences, parenthetical sentences have fewer limits, update quickly and are easy to obtain. As we can see in Example 1, the main task for extracting the correct bilingual term pairs is to find the left boundary of the target term. Most importantly, the bilingual term pairs in parenthetical sentences have greater quality compared to other text in various web pages.

In this paper, in order to achieve high accuracy, we propose a simple and effective framework to improve term translation by learning from parenthetical sentences. The proposed framework includes: (1) a focused web crawler, fetching and parsing relevant pages selectively; (2) a parenthetical sentence filter, acquiring parenthetical sentences including bilingual term pairs; (3) a term translation knowledge extractor, extracting bilingual term translation candidates; (4) a probability learner, generating the term translation table for MT decoders.

An overview of the proposed framework is shown in Figure 1. The input includes seed URLs and the seed dictionary. The final result is the term translation table with probabilities, being similar to phrase translation table in MT. In the processing flow, the intermediate results include the crawled web pages, extracted URLs, the filtered parenthetical sentences, the extracted incremental dictionary items and the extracted term translation candidate list. The key steps include identifying the left boundaries of target terms by employing a maximal entropy classifier, and generating the probabilities of items as shown in Example 2, in the term translation table in cooperating with SMT system. In this paper, we regard the term translation table as a feature of MT decoders.

During decoding in the sentence translation tasks, translation hypotheses are searched both in the phrase translation table and in the generated term translation table. The underlying MT decoder gets the scores of hypotheses from both tables, and selects the n-best list among translation hypotheses.

In the experiments, our proposed novel framework significantly improves the translation quality

Example 2: The term translation table based on Example 1

communication ||| 通信 ||| 0.387201 0.358436 0.623309 0.668845 ||| 0-0
interprocess ||| 间 ||| 0.00358423 0.0028275 0.333333 0.6 ||| 0-0
interprocess ||| 进程 间 ||| 0.333333 0.00160575 0.666667 0.24 ||| 0-0 0-1
interprocess communication ||| 间 通信 ||| 0.333333 0.000101348 0.333333 0.401307 ||| 0-0 1-1
interprocess communication ||| 进程 间 通信 ||| 0.4 0.000575558 0.666667 0.160523 ||| 0-0 0-1 1-2
IPC ||| 间 通信 ||| 0.333333 0.416858 0.0454545 0.352726 ||| 0-0 0-1
IPC ||| 进程 间 通信 ||| 0.65625 0.731707 0.5 0.120435 ||| 0-0 0-1 0-2

of terms and sentences. In summary, this paper makes the following contributions:

(1) The proposed simple and straightforward framework gets more reliable and accurate term translation knowledge by learning from parenthetical sentences. It substantially improves the translation quality of terms and sentences.

(2) The proposed framework regards the term translation table as a feature of MT decoders. It allows term translation knowledge to be more fully utilized compared with traditional bilingual term dictionaries.

(3) The well designed term translation knowledge extractor continuously extracts term translation candidates from parenthetical sentences. Some of the extracted candidates will be added into the seed dictionary as incremental dictionary items, so as to improve the accuracy of parenthetical sentences.

2 The Proposed Framework

In this section, we first introduce the whole framework, then give a detailed description of this framework in the following subsections.

The primary insight of the proposed framework is that authors of many mix-language web pages, especially non-English pages (such as Chinese, Japanese), usually annotate terms with their original English translations insides of a pair of parentheses. Thus we can extract some term translation pairs follow parenthesis pattern, especially for technical terms.

To achieve better term translation quality, our proposed framework includes four parts as shown in Figure 1:

(1) A focused web crawler, collecting relevant web pages. Different from general purpose crawlers, the crawler employed by this paper is a focused crawler, collecting web pages that contain parenthetical sentences. The proposed focused crawler will predict the probability that an unvisited page contains parenthetical sentences before actually downloading the page.

(2) A parenthetical sentence filter, acquiring parenthetical sentences including bilingual term pairs. There are various proposes of parenthesis patterns, such as term translation, explanation, supplement, examples. The proposed filter picks out sentences that contain only term translation and match the parenthesis pattern. Then parenthetical sentences will be acquired.

(3) A term translation knowledge extractor, extracting bilingual term translation candidates. The extractor identifies the left boundaries of target terms by employing a maximal entropy classifier. Then, the term translation candidate list for the parenthesized source term is extracted depending on the left anchor, namely the given left boundary. The classifier was trained by naturally annotated resources (e.g., Wikipedia) and the seed dictionary.

(4) A probability learner, generating the term translation table for MT decoders. Instead of extracting a multipurpose bilingual dictionary for many applications, in this paper, we design a probability learner to generate the term translation table with probabilities in cooperating with MT decoders. The learned probabilities help MT decoders achieve better translation quality compared with that of using bilingual term dictionary directly.

2.1 Focused Crawler

Crawlers used by general purpose search engines retrieve massive numbers of web pages regardless of their content. However, there are various kinds of web pages on the Internet, and only a small fraction of pages happens to contain parenthetical sentences. So, the focused crawler (Pal et al., 2009) is employed in this paper to collect targeted pages, by carefully prioritizing the crawl frontier and managing the hyperlink exploration process.

The proposed focused crawler in this paper will predict the probability that an unvisited page contains parenthetical sentences before actually downloading this page. The larger the probability, the higher the visiting priority will be assigned to the URL in the task queue.

A URL consists of the *domain*, the *path* and other parts. For instance, given the URL "https://en.wikipedia.org/wiki/Memory", the domain is "en.wikipedia.org" and the path is "/wiki/Memory". We assume that a page may contain more parenthetical sentences if: (1) other pages in the same domain have more parenthetical sentences; (2) the parent page from which the URL is extracted contains many parenthetical sentences. Therefore, the probability that a URL contains parenthetical sentences is calculated by:

$$\log p(url) = 0.5 \times \log \frac{count(url.domain)}{total(url.domain)} + 0.5 \times \log \frac{count(url.parent)}{total(url.parent)} \quad (1)$$

where $count$ refers to the number of parenthetical sentences, and $total$ refers to the number of sentences. The value of $count$ is given by the parenthetical sentence filter introduced in the next subsection. The focused crawler reorders the task queue by the probability according to Equation 1.

A Bloom filter is employed for filtering duplicate URLs, and the controlled Chromium browser is adopted to simulate keyboard and mouse actions for downloading pages which cannot be accessed in the general way.

2.2 Parenthetical Sentence Filter

There are various proposes of parenthesis patterns, such as term translation, explanation, supplement, and demonstration. Several typical parenthesis patterns are shown in Example 3. Only the patterns for term translation are focused in this paper, and other patterns should be eliminated.

In order to acquire parenthetical sentences for learning term translation, we design a parenthetical sentence filter to identify whether a sentence matching the parenthesis pattern should be retained or not. For a parenthetical sentence $s = c_1 c_2 \dots c_n (e_1 e_2 \dots e_m)$, the proposed filter combines the seed dictionary, co-occurrence and predefined rules to score the parenthetical sentence candidate according to the following equation:

$$\log p(s) = \lambda_1 \log p(domain) + \lambda_2 \log p(page) + \lambda_3 \log r(s) + \lambda_4 \log co(s) \quad (2)$$

In Equation 2, $r(s)$ refers to the ratio of source words that correspond to target words according to the dictionary can be matched before the left parenthesis and can be calculated by the following equations:

$$r(s) = \frac{1}{m} \times \sum_{j=1}^{m} sign(e_j) \quad (3)$$

$$sign(e_j) = \begin{cases} 0 & \forall\, t' \in dict(e_j),\ t' \notin \{c_n\} \\ 1 & \exists\, t' \in dict(e_j),\ t' \in \{c_n\} \end{cases} \quad (4)$$

where $dict(e_j)$ refers to the target word set of the source word e_j according to the seed dictionary.

In Equation 2, $co(s)$ denotes the average co-frequency of source words and target words and can be calculated by the following equation:

$$co(s) = \frac{1}{m} \times \sum_{j=1}^{m} \max_{1 \le i \le n, e_j \in s, c_i \in s} \frac{2 \times count(e_j, c_i)}{count(e_j) + count(c_i)} \quad (5)$$

After analysis, there are some typical websites and pages containing an especially great number of bilingual pairs. Such prior information is very helpful to recognize parenthetical sentences. Thus, in Equation 2, $s(domain)$ denotes the probability of one sentence included in $domain$ contains parenthetical term translation and can be derived as the following equation:

$$s(domain) = \frac{1}{|domain|} \sum_{s' \in domain} \left(\frac{\lambda_3 \times r(s')}{\lambda_3 + \lambda_4} + \frac{\lambda_4 \times co(s')}{\lambda_3 + \lambda_4} \right) \quad (6)$$

where $|domain|$ refers to the number of sentences in this domain. Similarly, the probability of one

Example 3: Several typical parenthesis patterns

Term translation:
软件开发中的**焦油坑**(the tar pit)可以通过尽责、专业的过程得以避免。
岩石里有种构造叫**夫妻节理**(英文：coupled joints)

Explanation:
蓟北：泛指蓟州、幽州一带(现在河北省北部地区)，是安、史叛军盘踞的地方。

Supplement:
艾米莉·狄金森(1830-1886)是美国文学史上一个伟大的诗人。
斯巴达克(杀开一条血路，大喊)不愿做奴隶的人们！起来！

Demonstration:
从图中两组节理面的**锐角**(beta)可计算出该岩石的内摩擦
转载请注意说明**来源**(www.qq.com)
没有被收录在词表中的词，包括各类**专有名词**(人名、地名、企业名等)

sentence included in $page$ contains parenthetical term translation, $s(page)$, can be derived as the following equation:

$$s(page) = \frac{1}{|page|} \sum_{s' \in page} \left(\frac{\lambda_3 \times r(s')}{\lambda_3 + \lambda_4} + \frac{\lambda_4 \times co(s')}{\lambda_3 + \lambda_4} \right) \quad (7)$$

where $|page|$ refers to the number of sentences in this page.

In this paper, the default values of λ are set to the following weights: $\lambda_1 = \lambda_2 = 0.2$, $\lambda_3 = \lambda_4 = 0.3$.

2.3 Term Translation Knowledge Extractor

In order to extract bilingual term translation candidates, the key task is to identify the left boundary of a target term. However, traditional term recognition methods employing statistical measures to rand the candidates terms (n-gram sequences), such as log likelihood (Cohen, 1995; Lefever et al., 2009), TF-IDF (Evans and Lefferts, 1995; Medelyan and Witten, 2006), C-value/NC-value (Frantzi et al., 2000) and many others (Ahmad et al., 2000; Park et al., 2002; Kozakov et al., 2004; Sclano and Velardi, 2007; Zhou et al., 2008; Zhang et al., 2008; Kostoff et al., 2009), leads to very low recall for some domains. What's worse, some approaches apply frequency threshold to reduce the algorithm's search space by filtering out low frequency term candidates. Such methods have not taken into account Zipf's law, again leading to the reduced recall.

In this paper, to achieve a higher recall, we adopt naturally annotated resources for term recognition and focus on supervised machine learning approaches based recognition approaches for MT with a wide range of fields. Thus, we train a maximal entropy based term recognition model using Wikipedia sentences to detect the left boundary candidate of a given target term.

There are plenty of naturally annotated resources with parenthetical sentences that can be used to train the term recognizer as shown in Figure 2, especially Wikipedia. In Figure 2, the phrases with red rectangles are terms. As we can see, this terms are naturally annotated with bold fonts or hyperlinks. And such natural annotations clearly provide the important boundary information of terms and can be adopted as training data of term recognizers.

In this paper, we design following features for the term recognizer: the four words immediately to the left of the term, $W_{s-4}, \ldots, W_{s-1}$, and their parts of speech, $POS_{s-4}, \ldots, POS_{s-1}$; the four words immediately to the right of the term $W_{s+1}, \ldots, W_{s+4}$, and their parts of speech, $POS_{s+1}, \ldots, POS_{s+4}$; the first word of the phrase WL and the part of speech $POSL$; the last word of the phrase WR and the part of speech $POSR$; the ratio of target words, D, that match parenthetical source words according to the seed dictionary.

In this way, we can get the probability $p(c_i)$ of an anchor, the first word of the term. Then, the term translation candidate list for the parenthesized source term is extracted depending on the left anchor. An example of extracted English-Chinese term translation candidates is shown in Table 1.

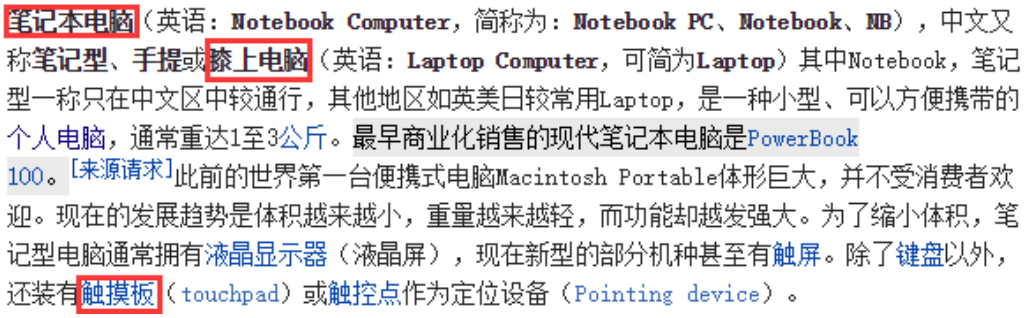

Figure 2: Naturally annotated resources.

To expand the seed dictionary, the items with high confidence in the term translation candidate list will be selected as incremental dictionary items. By doing this, we can make up for the seed dictionary as the growth of term pairs.

2.4 Probability Learner

In order to substantially improve the quality of term and sentence translation, in this paper, we design a probability learner to generate the term translation table with probabilities in cooperating with SMT decoders. The probability learner combines word alignment model with the detected boundary candidates to generate the final term translation table. And the process of searching for the best boundary, c_i, can be formulated as the joint model:

$$i = \underset{1 \le i \le n}{\operatorname{argmax}}\, p(c_i)^{\lambda_5} \times p(c_i \dots c_n|e)^{\lambda_6} \times p(e|c_i \dots c_n)^{\lambda_7} \quad (8)$$

where $p(c_i \dots c_n|e)$ and $p(e|c_i \dots c_n)$ are word alignment probabilities of the source term and the target term, and $e = e_1 \dots e_m$.

In Example 1, s = "所以 只能 使用 进程 间 通讯 (interprocess communication , IPC)". For the source term "interprocess communication", c_1 = "所以", c_2 = "只能", c_3 = "使用", c_4 = "进程", c_5 = "间", c_6 = "通讯", e_1 = "interprocess", e_2 = "communication". And the left boundary is incorrectly recognized by our baseline system as c_5, namely, the target term is c_5c_6 ="间 通讯". In order to correct the detection error, we enlarge or shrink the anchor from the left boundary to re-generate target terms, including the correct target term $c_4c_5c_6$ ="进程 间 通讯". Then, we select a best regenerated term which maximizes the joint probability according to Equation 8. In this work, the HMM-based word alignment model (Vogel et al., 1996) is employed to align words.

Next, we can extract term translation rules using the selected term above, and generate the term translation table as shown in Example 2. In Example 2, fields of the line "communication ||| 通信 ||| 0.387201 0.358436 0.623309 0.668845 ||| 0-0" includes 7 properties: source term, target term, phrase translation probability, lexical weights, inverse phrase translation probability, inverse lexical weights, word alignment.

In this paper, the default values of λ_5, λ_6 and λ_7 are set to 0.4, 0.3 and 0.3, respectively.

3 Experiments

We conduct the experiments to test the performance of our proposed framework on improving the quality of term and sentence translation. We will check how much improvement the proposed framework can achieve on the final MT results. The performance of term pair extraction is evaluated by precision (P); the quality of term translation and sentence translation are evaluated by precision (P) and BLEU, respectively.

3.1 Experimental Setup

All the experiments are conducted on our in-house developed MT toolkit which has a typical phrase-based decoder (Xiong et al., 2006) and a series of tools, including word alignment and phrase table extraction.

We test our method on English-to-Chinese translation in the field of software localization. The training data (1,199,589 sentences) and annotated test data (1,100 sentences) are taken from Microsoft Translation Memory, which is a domain-specific dataset. And additional data employed by this paper includes: the seed dictionary (102,308 source words[2], 24,094 terms from Mi-

[2]http://www.mdbg.net/chindict/chindict.php?page=cc-cedict

Source	Target
Mihr-Ohrmazd	拂多诞
Wicca	威卡尔
Francis Dashwood	弗朗西斯达希武德
Religious Studies	宗教学
Introduction to the Science of Religion	宗教科学引论
History of Religions	宗教史学
Phenomenology of Religion	宗教现象学
anomalous monism	无法则一元论
qualia	感质
Panspermia	泛种论
Determinism	决定论

Table 1: Extracted English-Chinese term translation candidates

crosoft Terminology Collection), Chinese Pinyin table (7,809 Chinese characters[3]). The gold standard of term translation of test data are human annotated.

All the MT systems are tuned by the development set (1,000 sentences) using ZMERT (Zaidan, 2009) with the objective to optimize BLEU (Papineni et al., 2002). The higher the BLEU score, the better the translation is. And the statistical significance test is performed by the re-sampling approach (Koehn, 2004).

3.2 Results and Analysis

(1) The Term Extraction Tests

Firstly, we will evaluate the extraction performance of term translation candidates. In our experiments, the focused crawler has downloaded 162,543,832 web pages. And there are 12,673,286 pages that contain 49,976,931 parenthesized sentences selected by the parenthesized sentence filter in total.

The baseline term extraction system is denoted as "Baseline" which is implemented according to the work introduced by (Cao et al., 2007). The baseline system has extracted 10,823,132 terms from above web pages. And our system, being denoted as "TermExt", outputs 12,048,310 terms. As we can see, the **recall** has been increased by 11.32% using our proposed framework.

Then, We sample the extracted terms 10 times on the baseline system and the proposed framework, respectively. And each sample includes 1,000 terms. And we report the average precision in Table 2.

[3]http://www.51windows.net/pages/gb2312.htm

In contrast to the baseline approach, the figures in Table 2 show that the **precision** of Chinese terms has been increased by 2.9 points, and the **precision** of term pairs has been increased by 4.1 points. Thus, according to the bold figures in Table 2, we can draw a conclusion that term extraction can be substantially increased by the proposed framework.

(2) The SMT Translation Tests

Secondly, we test whether the proposed framework can further improve the performance of term and sentence translation, compared with the baseline system. The strong baseline system, e.g., well tuned Moses, is denoted as "Moses". And our SMT system is denoted as "MaxEntSMT". The translation results based on the extracted term dictionary are labeled with "MaxEntSMT+BaselineDict" and "MaxEntSMT+TermExtDict", respectively. Correspondingly, the translation results based on the term translation table are labeled with "MaxEntSMT+TermExtTable". The word alignment was conducted bidirectionally and then symmetrized for extracting phrases as Moses (Koehn et al., 2007) does. The test set includes 1,100 sentences with 1,208 bilingual term pairs altogether. In order to highlight the performance of term translation, we count the number of terms that are translated exactly correctly, and the corresponding term translation results are denoted as "Term (P/%)" (exactly matching). Meanwhile, the sentence translation results are labeled "Sentence (BLEU/%)". We report all the results in Table 3.

In Table 3, our in-house developed SMT system makes the translation result worse than Moses.

	Number of Terms	Chinese Terms (P/%)	Term Pairs (P%)
Baseline	10,823,132	94.30	88.20
TermExt	12,048,310	**97.20****	**92.30****

"**" means the scores are significantly better than previous lines with $p < 0.01$.

Table 2: The performance of term extraction

	Term (P/%)	Sentence (BLEU/%)
Moses	86.43	46.01
MaxEntSMT	86.14	45.93
MaxEntSMT+BaselineDict	89.47	46.19
MaxEntSMT+TermExtDict	91.22	46.35
MaxEntSMT+TermExtTable	**94.38****	**47.26****

"**" means the scores are significantly better than previous lines with $p < 0.01$.

Table 3: The performance of translation

However, with the help of the proposed framework, the term translation quality is significantly improved by more than 7.95% accuracy. Non-term words are also strongly improved by the framework, because that the accuracy ratio of term words translation has been much improved and fewer non-term words are translated incorrectly. In sentence translation, the bold figures in Table 3 demonstrate that it improves the translation quality by 1.25 absolute BLEU points, compared with the well tuned Moses. Considering one term on average in a single sentence in the test set, the BLEU scores are very promising actually, and our goals on term translation have been achieved.

In summary, we can draw the conclusion that the proposed term extraction framework significantly improves the performance of term extraction from web pages, and further substantially improves the performance of MT in term of term translation and sentence translation.

4 Related Work

For term translation pairs extraction from parenthetical sentences, Cao *et al.*(Cao et al., 2007) and Lin *et al.*, like us, proposed two different methods to mine bilingual dictionaries. Cao *et al.* used a 300GB collection of web documents as input. They extracted candidate translations using predefined templates, and used supervised learning to build models that deal with phonetic transliterations and semantic translations separately. Lin *et al.* used a word alignment algorithm, not to make a distinction between translations and transliterations, to identify the terms being translated relying on unsupervised learning.

Our work depends on supervised learning with naturally annotated resources (e.g., Wikipedia) to train a term recognization model and detect left boundary candidates of a term. In addition, our method combines word alignment model with the detected boundary candidates to generate the final term translation table with probabilities for MT, rather than the extracted bilingual term dictionary. We make no distinction between translations and transliterations.

5 Conclusion

In this paper, we have presented an simple, straightforward and effective framework to learn from parenthetical sentences for term translation in MT. The proposed framework continuously extracts term translation candidates from parenthetical sentences from web pages, generates the term translation table, then regards the term translation table as a feature of MT decoders, finally substantially boosts term translation and sentence translation. The experimental results are promising.

References

Khurshid Ahmad, Lee Gillam, and Lena Tostevin. 2000. Weirdness indexing for logical document extrapolation and retrieval. In *Proceedings of the Eighth Text Retrieval Conference (TREC-8)*.

Guihong Cao, Jianfeng Gao, Jian-Yun Nie, and WA Redmond. 2007. A system to mine large-scale bilingual dictionaries from monolingual web. *Proceedings of MT Summit XI*, pages 57–64.

Jonathan D. Cohen. 1995. Highlights: Language- and domain-independent automatic indexing terms for abstracting. *Journal of the American Society for Information Science*, 46(3):162–174.

David A. Evans and Robert G. Lefferts. 1995. Clarittrec experiments. *Information Processing and Management*, 31(3):385–395.

Katerina Frantzi, Sophia Ananiadou, and Hideki Mima. 2000. Automatic recognition of multi-word terms: the c-value/nc-value method. *International Journal on Digital Libraries*, 3(2):115–130.

Philipp Koehn. 2004. Statistical significance tests for machine translation evaluation. In *Proceedings of EMNLP 2004*.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, and Richard Zens. 2007. Moses: open source toolkit for statistical machine translation. In *Proceedings of ACL 2007*.

Ronald N. Kostoff, Joel A. Block, Jeffrey L. Solka, Michael B. Briggs, Robert L. Rushenberg, Jesse A. Stump, Dustin Johnson, Terence J. Lyons, and Jeffrey R. Wyatt. 2009. Literature-related discovery. *Annual Review of Information Science and Technology*, 43(1):171.

L Kozakov, Y. Park, T. H. Fin, Y. Drissi, Y N Doganata, and T. Cofino. 2004. Glossary extraction and knowledge in large organisations via semantic web technologies. In *Proceedings of the 6th International Semantic Web Conference and the 2nd Asian Semantic Web Conference*.

Els Lefever, Lieve Macken, and Veronique Hoste. 2009. Language-independent bilingual terminology extraction from a multilingual parallel corpus. In *Proceedings of EACL 2009*.

Olena Medelyan and Ian H. Witten. 2006. Thesaurus based automatic keyphrase indexing. In *Proceedings of the ACM/IEEE-CS Joint Conference on Digital Libraries*.

Anshika Pal, Deepak Singh Tomar, and SC Shrivastava. 2009. Effective focused crawling based on content and link structure analysis. *arXiv preprint arXiv:0906.5034*.

Kishore Papineni, Salim Roukos, Todd Ward, and Weijing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of ACL 2002*.

Youngia Park, Roy J Byrd, and Branimir K Boguraev. 2002. Automatic glossary extraction: beyond terminology identification. In *Proceedings of COLING 2002*.

Feiliang Ren, Jingbo Zhu, and Huizhen Wang. 2010. Web-based technical term translation pairs mining for patent document translation. In *Proceedings of the 6th International Conference on Natural Language Processing and Knowledge Engineering (NLP-KE 2010)*.

F. Sclano and P. Velardi. 2007. Termextractor: a web application to learn the shared terminology of emergent web communities. In *Proceedings of the 3rd International Conference on Interoperability for Enterprise Software and Applications (I-ESA 2007)*.

Stephan Vogel, Hermann Ney, and Christoph Tillmann. 1996. Hmm-based word alignment in statistical translation. In *Proceedings of the 16th conference on Computational linguistics*, volume 2, pages 836–841.

Deyi Xiong, Qun Liu, and Shouxun Lin. 2006. Maximum entropy based phrase reordering model for statistical machine translation. In *proceedings of COLING-ACL 2006*.

Omar F. Zaidan. 2009. Z-mert: A fully configurable open source tool for minimum error rate training of machine translation systems. *The Prague Bulletin of Mathematical Linguistics*, 91:79–88.

Ziqi Zhang, J. Iria, and Christopher Brewster. 2008. A comparative evaluation of term recognition algorithms. In *LREC 2008*.

Zili Zhou, Yanna Wang, and Junzhong Gu. 2008. Markov-based automatic term extraction. In *Future BioMedical Information Engineering, 2008*.