Large-scale Japanese-Chinese Scientific Dictionary Construction via Pivot-based Statistical Machine Translation

Chenhui Chu[1,3], **Raj Dabre**[1], **Toshiaki Nakazawa**[1,2] **and Sadao Kurohashi**[1]
[1]Graduate School of Informatics, Kyoto University
[2]Japan Science and Technology Agency
[3]Japan Society for the Promotion of Science Research Fellow
{chu,raj,kuro}@nlp.ist.i.kyoto-u.ac.jp nakazawa@pa.jst.jp

1 Introduction

Pivot-based statistical machine translation (SMT) (Wu and Wang, 2007) has been shown a possible way of constructing a dictionary for the language pairs that have scarce parallel data (Tsunakawa et al., 2009). The assumption of this method is that there is a pair of large-scale parallel data: one between the source language and an intermediate resource rich language (henceforth called pivot), and one between that pivot and the target language. Once this assumption suffices, we can use the source-pivot and pivot-target parallel data to develop a source-target term[1] translation model for dictionary construction.

There are two main advantages of pivot-based SMT for dictionary construction. One is that because pivot-based SMT uses the log linear model as conventional phrase-based SMT (Koehn et al., 2007) does, various features can be integrated to improve the accuracy. The other is that this method can address the data sparseness problem of directly merging the source-pivot and pivot-target terms, because it can use the portion of terms to generate new terms. However, the potential of this method has not been fully explored. Small-scale experiments in (Tsunakawa et al., 2009) showed very low accuracy of pivot-based SMT for dictionary construction.[2]

This paper presents our study to construct a large-scale Japanese-Chinese (Ja-Zh) scientific dictionary, using large-scale Japanese-English (Ja-En) ($49.1M$ sentences and $1.4M$ terms) and English-Chinese (En-Zh) ($8.7M$ sentences and $4.5M$ terms) parallel data via pivot-based SMT. We generate a large pivot translation model using the Ja-En and En-Zh parallel data. Moreover, a small direct Ja-Zh translation model is generated using small-scale Ja-Zh parallel data ($680k$ sentences and $561k$ terms). Both the direct and pivot translation models are used to translate the Ja terms in the Ja-En model to Zh, to construct a large-scale Ja-Zh dictionary (about $58.5M$ terms). In addition, we exploit linguistic knowledge of common Character characters (Chu et al., 2013) shared in Ja-Zh to further improve the translation model. Large-scale experiments on scientific domain data indicate that dictionary construction via pivot-based SMT can achieve high enough accuracy for practical use.

2 Phrase-based SMT

This section gives a brief overview of phrase-based SMT (Koehn et al., 2007), which is the foundation of pivot-based SMT. In phrase-based SMT, the translation model is represented as a phrase table, containing phrase pairs together with their feature scores. The phrase pairs are extracted from a parallel corpus based on unsupervised word alignments. Inverse and direct phrase translation probabilities $\phi(f|e)$ and $\phi(e|f)$, inverse and direct lexical weighting $lex(f|e)$ and $lex(e|f)$ are used as features for the phrase table. Phrase translation probabilities are calculated via maximum likelihood estimation, which counts how often a source phrase f is aligned to target phrase e in the parallel corpus, and vise versa. Lexical weighting is the average word translation probability calculated using internal word align-

[1] In this paper, we call the entries in the dictionary terms. A term consists of one or multiple tokens.

[2] The highest accuracy evaluated based on the 1 best translation is 0.217 in (Tsunakawa et al., 2009).

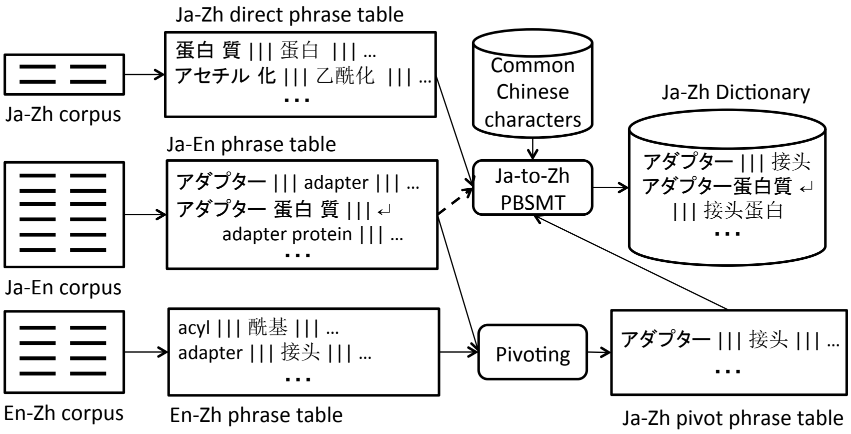

Figure 1: Overview of our dictionary construction method.

ments of a phrase pair, which is used to smooth the overestimation of the phrase translation probabilities. Other typical features such as the reordering model features and the n-gram language model features are also used in phrase-based SMT. These features are combined in a log linear model, and their weights are tuned using a small size of parallel sentences. During decoding, these features together with their tuned weights are used to produce new translations.

3 Dictionary Construction via Pivot-based SMT

Figure 1 illustrates an overview of our construction method. We first generate Ja-Zh (source-target), Ja-En (source-pivot) and En-Zh (pivot-target) phrase tables from parallel data respectively. The generated Ja-Zh phrase table is used as the direct table. Using the Ja-En and En-Zh phrase tables, we construct a Ja-Zh pivot phrase table via En. The direct and pivot tables are then combined and used for phrase-based SMT to translate the Ja phrases in the Ja-En phrase table to Zh, to construct a large-scale Ja-Zh dictionary. In addition, we use common Chinese characters to generate Chinese character features for the phrase tables to improve the SMT performance.

3.1 Pivot Phrase Table Generation

We follow the phrase table triangulation method (Wu and Wang, 2007) to generate the pivot phrase table. This method generates a source-target phrase table via all their shared pivot phrases in the source-pivot and pivot-target tables. The formulae for generating the inverse and direct phrase translation probabilities $\phi(f|e)$ and $\phi(e|f)$, inverse and direct lexical weighting $lex(f|e)$ and $lex(e|f)$ for the generated source-target phrase pairs using the pivots are:

$$\phi(f|e) = \sum_{p_i} \phi(f|p_i) * \phi(p_i|e) \tag{1}$$

$$\phi(e|f) = \sum_{p_i} \phi(e|p_i) * \phi(p_i|f) \tag{2}$$

$$lex(f|e,a) = \sum_{p_i} lex(f|p_i,a_1) * lex(p_i|e,a_2) \tag{3}$$

$$lex(e|f,a) = \sum_{p_i} lex(e|p_i,a_2) * lex(p_i|f,a_1) \tag{4}$$

where a_1 is the alignment between phrases f (source) and p_i (pivot), a_2 is the alignment between p_i and e (target) and a is the alignment between e and f.

3.2 Combination of the Direct and Pivot Phrase Tables

To combine the direct and pivot phrase tables, we make use of the multiple decoding paths (MDP) method of the phrase-based SMT toolkit Moses (Koehn et al., 2007). MDP uses both the tables simultaneously while decoding. Translation options are collected from both the tables, which allows for more translation options.

3.3 Chinese Character Features

Ja-Zh shares Chinese characters. Because many common Chinese characters exist in Ja-Zh, they have been shown to be very effective in many Ja-Zh natural language processing (NLP) tasks (Chu et al., 2013). In this paper, we compute Chinese character features for the phrase pairs in the translation models, and integrate these features in the log-linear model for decoding. In detail, we compute following two features for each phrase pair:

$$CC\_ratio = \frac{Ja\_CC\_num + Zh\_CC\_num}{Ja\_char\_num + Zh\_char\_num} \quad (5)$$

$$CCC\_ratio = \frac{Ja\_CCC\_num + Zh\_CCC\_num}{Ja\_CC\_num + Zh\_CC\_num} \quad (6)$$

where $char\_num$, CC_num and CCC_num denote the number of characters, Chinese characters and common Chinese characters in a phrase respectively. The common Chinese character ratio is calculated based on the Chinese character mapping table in (Chu et al., 2013).

4 Experiments

We conducted experiments to show the effectiveness of our dictionary construction method. The accuracy of our method was evaluated using manually created dictionaries.

4.1 Data

4.1.1 Training data

We used following two types of training data:

- Bilingual dictionaries: we used the scientific Ja-En, En-Zh and Ja-Zh dictionaries provided by the Japan Science and Technology Agency (JST)[3] and the Institute of Science and Technology information of China (ISTIC),[4] containing $1.4M$, $4.5M$ and $561k$ term pairs respectively.

- Parallel corpora: the scientific Ja-En, En-Zh and Ja-Zh corpora we used were also provided by JST and ISTIC, containing $49.1M$, $8.7M$ and $680k$ sentence pairs respectively.

4.1.2 Tuning and Testing data

We used the terms with two reference translations[5] in the Ja-Zh Iwanami biology dictionary (5,890 pairs) and the Ja-Zh life science dictionary (4,075 pairs) provided by JST. Half of the data in each dictionary was used for tuning (4,983 pairs), and the other half for testing (4,982 pairs).

4.2 Settings

We compared following two training data settings:

- Dic: Only use the dictionaries for training.

- Corpus+Dic: Use both the dictionaries and corpora for training.

In addition, we compared following three methods for training the translation model:

- Direct: Only use the Ja-Zh data to train a direct Ja-Zh model.

- Pivot: Use the Ja-En and En-Zh data for training Ja-En and En-Zh models, and construct a pivot Ja-Zh model using the phrase table triangulation method.

- Direct+Pivot: Combine the direct and pivot Ja-Zh models using MDP.

For the models trained on Corpus+Dic, we also conducted experiments additionally using the Chinese character features (labeled +CC). For decoding, we used Moses (Koehn et al., 2007) with the default options. We trained a word 5-gram language model on the Zh side of all the En-Zh and Ja-Zh training data ($14.4M$ sentences) using the SRILM toolkit[6] with interpolated Kneser-Ney discounting. Tuning was performed by minimum error rate training, and it was re-run for every experiment.

4.3 Evaluation Criteria

Following (Tsunakawa et al., 2009), we evaluated the accuracy on the test set using the following three metrics:

- 1 best: Percentage of terms where the top 1 translation given by the MT system is the correct one.

- 20 best: Percentage of terms where the correct translation is included in the top 20 translations given by the MT system.

- Mean Reciprocal Rank (MRR): Let w be a source term, $rank_w$ denotes the rank of its correct translation within the list of translations given by the MT system, V denotes the set of

[3] http://www.jst.go.jp

[4] http://www.istic.ac.cn

[5] Different terms are annotated with different number of reference translations in these two dictionaries.

[6] http://www.speech.sri.com/projects/srilm

				Accuracy w/ OOV			Accuracy w/o OOV		
Training data	Method	BLEU-4	OOV term	1 best	20 best	MRR	1 best	20 best	MRR
Dic	Direct	31.22	37%	0.2912	0.3792	0.3239	0.4476	0.5833	0.4978
	Pivot	45.32	18%	0.4123	0.6202	0.4836	0.5022	0.7559	0.5893
	Direct+Pivot	46.24	18%	0.4213	0.6204	0.4925	0.5115	0.7542	0.5983
Corpus+Dic	Direct	40.64	26%	0.3697	0.5255	0.4258	0.4978	0.7082	0.5736
	Direct+CC	40.84	26%	0.3721	0.5255	0.4271	0.5011	0.7082	0.5754
	Pivot	52.11	9%	0.4914	0.7252	0.5712	0.5371	0.7927	0.6243
	Pivot+CC	53.24	9%	0.4984	0.7258	0.5766	0.5448	0.7933	0.6302
	Direct+Pivot	53.54	8%	0.5024	0.7377	0.5875	0.5485	0.8054	0.6414
	Direct+Pivot+CC	54.17	8%	0.5157	0.7356	0.5950	0.5630	0.8032	0.6496

Table 1: Evaluation results.

terms used for evaluation. Then MRR is defined as:

$$MRR = \frac{1}{|V|} \sum_{w \in V} \frac{1}{rank_w} \qquad (7)$$

We used the top 20 translations given by the MT system for calculating MRR.

In addition, we report the BLEU-4 scores that were computed on the word level.

4.4 Results

Table 1 shows the evaluation results. We also show the percentage of out-of-vocabulary (OOV) terms,[7] and the accuracy with and without OOV terms respectively. In general, we can see that Pivot performs better than Direct, because the data of Ja-En and En-Zh is larger than that of Ja-Zh. Direct+Pivot shows better performance than either method. Training on more data of Corpus+Dic is better than Dic only. Chinese character features can further improve the accuracy.

However, the accuracy of our best performing method Direct+Pivot+CC (Corpus+Dic) is still not high enough according to our evaluation method. We manually investigated the terms, whose top 1 translation was evaluated as incorrect according to our evaluation method. Based on our investigation, nearly 75% of them were actually correct translations. They were undervalued because they are not covered by the reference translations in our test set. Taking this observation into consideration, the actual 1 best accuracy is about 0.9. Automatic evaluation tends to greatly underestimate the results because of the incompleteness of the test set.

As there are $636M$ unique Ja phrases in the Ja-En Corpus+Dic phrase table and we suppose that 10% of them are terms, we are able to construct a large-scale Ja-Zh dictionary containing $58.5M$[8] terms.

5 Conclusion and Future Work

In this paper, we presented a dictionary construction method via pivot-based SMT. Large-scale Ja-Zh experiments showed the effectiveness of our method. In the future, we plan to further improve the accuracy of our method by significance testing based pruning and integrating contextual features estimated on the source and target corpora.

References

Chenhui Chu, Toshiaki Nakazawa, Daisuke Kawahara, and Sadao Kurohashi. 2013. Chinese-japanese machine translation exploiting chinese characters. *ACM Transactions on Asian Language Information Processing (TALIP)*, 12(4):16:1–16:25.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondrej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: Open source toolkit for statistical machine translation. In *Proceedings of ACL*, pages 177–180.

Takashi Tsunakawa, Naoaki Okazaki, Xiao Liu, and Jun'ichi Tsujii. 2009. A chinese-japanese lexical machine translation through a pivot language. *ACM Transactions on Asian Language Information Processing (TALIP)*, 8(2):9:1–9:21, May.

Hua Wu and Haifeng Wang. 2007. Pivot language approach for phrase-based statistical machine translation. *Machine Translation*, 21(3):165–181, September.

[7] An OOV term contains at least one OOV word.

[8] $636M * 10\% * (1 - 8\%) = 58.5M$