高精度の単語アライメントを利用したSMTベース低資源機械翻訳

Using Highly Accurate Word Alignment for Low Resource Translation based on SMT

Division of Intelligent Interaction Technologies, Faculty of Engineering, Information and Systems, University of Tsukuba ／筑波大学システム情報系知能機能工学域教授

Takehito Utsuro ／宇津呂　武仁

Takehito Utsuro is a professor at the Division of Intelligent Interaction Technologies, Faculty of Engineering, Information and Systems, University of Tsukuba, since 2012. His professional interests in natural language processing, Web intelligence, information retrieval, machine learning, spoken language processing, and artificial intelligence.

Master's Program in Intelligent and Mechanical Interaction Systems, Degree Programs in Systems and Information Engineering, Graduate School of Science and Technology, University of Tsukuba ／筑波大学大学院理工情報生命学術院システム情報工学研究群知能機能システム学位プログラム

Jingyi Zhu ／朱　鏡伊

Jingyi Zhu is a student of Master's Program in Intelligent and Mechanical Interaction Systems, Degree Programs in Systems and Information Engineering, Graduate School of Science and Technology, University of Tsukuba.

Master's Program in Intelligent and Mechanical Interaction Systems, Degree Programs in Systems and Information Engineering, Graduate School of Science and Technology, University of Tsukuba ／筑波大学大学院理工情報生命学術院システム情報工学研究群知能機能システム学位プログラム

Takuya Tamura ／田村　拓也

Takuya Tamura is a student of Master's Program in Intelligent and Mechanical Interaction Systems, Degree Programs in Systems and Information Engineering, Graduate School of Science and Technology, University of Tsukuba.

NTT Communication Science Laboratories, NTT Corporation ／NTT コミュニケーション科学基礎研究所

Masaaki Nagata ／永田　昌明

Masaaki Nagata is a senior distinguished researcher at NTT Communication Science Laboratories, NTT Corporation, since 2012. His professional interests in machine translation and word segmentation.

1 Introduction

Neural Machine Translation (NMT), a data-hungry technology, suffers from the lack of bilingual data in low resource constraints[17][20][4]. To overcome the challenge that exists in small-scale translation or low resource translation tasks, several kinds of research focus on the approaches such as pre-training with large scale monolingual data and fine-tuning with a small-scale corpus[10][19], or using the unsupervised methods and mapping the monolingual vector embeddings into a common cross-lingual embedding space[9][16]. However, these effective methods need relatively high computation[10].

In this paper, we proposed a framework based on SMT and highly accurate word alignment methods SpanAlign and AWESoME-align, to explore the feasibility of low resource language translation. We use both the original order sentences and pre-ordered sentences as input because, with the help of high precision word alignment, it is hard to predict which side as input will be translated in a better result. Since

we focus on the limitation of the low resource language pairs, we use the Asian Language Tree bank corpus, which contains 20,000 parallel sentences as the base corpus. We do the experiments between the directions of the Japanese, Chinese, English, Bengali, Filipino, Hindi, Indonesian, Malay, and Thai, while Japanese is either the source side or the target side. The experimental results show that our proposed framework significantly outperforms an NMT based on the Transformer-base, and except for Ja-Th, the best results of each language pairs outperformed Transformer-small.

2 Related Work

Although preordering has often been used in SMT related works, some studies have recently applied preordering to NMT. Kawara et al.[7] discusses the influence of word order on the NMT model and concludes that it is important to keep the consistency between the input source word order and the output target word order, to improve the translation accuracy. Murthy et al.[11] proposed a transfer learning approach for NMT, that trains an NMT model on an assisting language-target language pair, and improves the translation quality in extremely low-resource scenarios. Nevertheless, those methods both rely on the neural network translation model or separately pre-training a translation model by a large-scale corpus. In contrast, our proposed framework has no neural translation component and we focus on the translation task limited to a small-scale corpus.

3 The Framework Based on SMT and Word Alignment

As shown in Figure 1(a), for the beginning of the process, we fine-tune the multilingual BERT-based word aligner using the manually made word alignment data or parallel corpus. Then we use the word alignment model to align words in the training sentences, while the word alignment data is used to train the Moses model, consisting of the phrase table[1] and statistical-based language model. At last, the original order test data or preordered test data is translated by the phrase-based SMT model. On the other hand, Figure 1(b) shows the procedure to create preordered test data.

The word alignment of the training corpus is also used to train the Pointer Network. Then the trained Pointer Network transforms the original order test data into preordered test data.

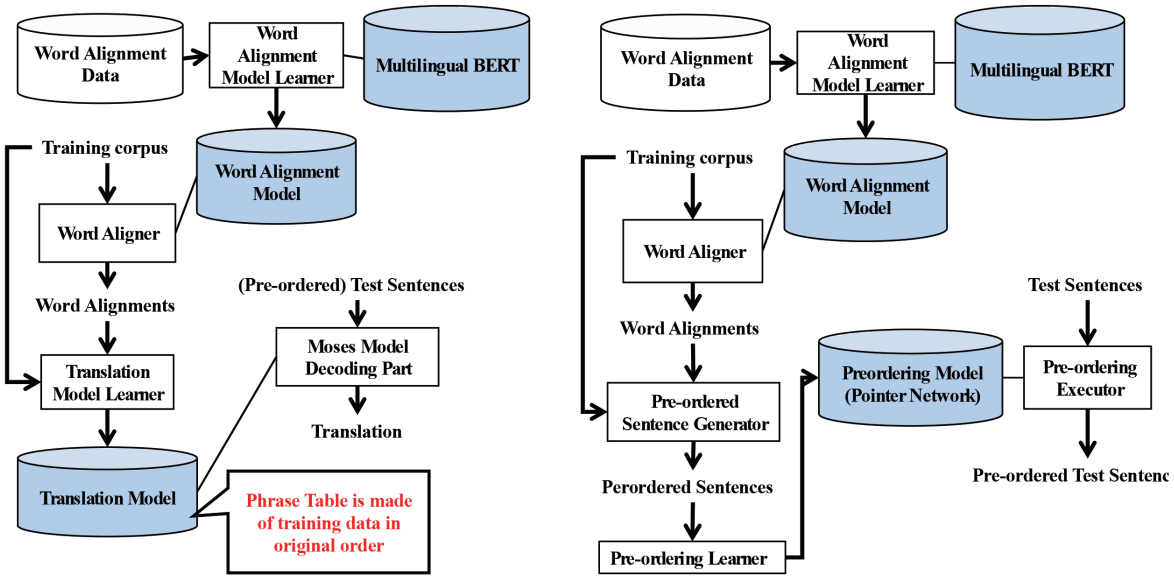
3.1 Multilingual BERT-based Word Aligners

The first method is SpanAlign [12], which extracts alignments with reading comprehension style, that inputs with the source language sentence and target language sentence, and predicts a span in the target sentence corresponding to the word in the source sentence enclosed between the two boundary symbols. This approach allows for high precision alignment even with less word alignment fine-tuning data in a supervised way.

The second method is AWESoME-align[3], which, on the other side, can be fine-tuned in an unsupervised way by adjusting the embedding distribution of mBERT output to achieve word alignments. The advantage is that this method does not require manually made word alignment data.

In our experiments, we fine-tune SpanAlign

1 Unlike the conventional pre-ordering translation, in our case, the phrase table is made in the original order, which means, we did not use the pre-ordered sentences to learn the probability of the phrase table. We also tried making the phrase table after pre-ordering the training data, however, the translation BLEU score is lower than that made by the original order data.



(a) Overall Flow of Accepting the Original Order Pre-ordered Sentence as Translation Input

(b) Using Pointer Network in Pre-ordering Sentence or

Figure 1: Our proposed framework.

and AWESoME-align respectively and compare them with the unfine-tuned (directly using parameters of pre-trained mBERT) AWESoME-align. Because of the supervised attribute of SpanAlign, we only leverage it on Ja-En and Ja-Zh pairs.

3.2 Pre-ordering by Pointer Network

3.2.1 Architecture

The pre-ordering process transforms the orders of the tokens in a source sentence to those of the tokens in its target sentence before translation is performed. Figure 3 shows an example of transferring a Japanese sentence.

The original Pointer Network[18] is an LSTM[5] based neural network, which aims at solving graph theory problems such as the traveling salesman problem and convex hull. Structurally, an encoding RNN converts the input sequence to a vector that is fed to the generating network. And at each step, the generating network produces a vector that modulates a content-based attention mechanism over inputs.

The output of the attention mechanism is a softmax distribution with a dictionary size equal to the length of the input.

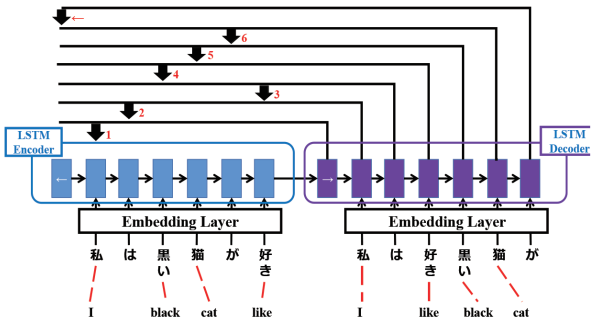


Figure 2: Architecture of Pointer Network for sequence order transformation (the modified Pointer Network accepts the original order sequence as input, and outputs the pre-ordered sequence).

Inspired by this, we apply Pointer Network to word order rearrangement like Figure 2. Specifically, we replace the input of Pointer Network with a sequence of the token instead, and then add an embedding layer to represent words with vectors. At decoding time, the decoder predicts the next pointer probability $p(C_i \mid C_1, \ldots, C_{i-1}, P)$ relying on inputs and predicted outputs :

$$u_j^i = v^T \tanh(W_1 e_j + W_2 d_i) \quad j = 1,\ldots, n$$
$$p(C_i \mid C_1,\ldots,C_{i\ 1}, P) = \text{softmax}(u^i)$$

where softmax normalizes the vector (of length n) to be an output distribution of inputs. P is the input sentence, and C_i is the token of the output sentence, u^i is the vector. Parameters v, W_1, are learnable parameters of the output model, and e_j , d_i represent the encoder state and decoder state, respectively.

3.2.2 Phrase-based Translation

Phrase-based SMT (PSMT) is found more efficient than word-based SMT framework thanks to the use of multi word translation units [1], and for the translation part, the phrase table plays a significant role. Bilingual phrase tables can be simply seen as lists of terms (words or phrases) in one language associated with their translations in a second language. Therefore, Phrase-based translation is a process that, for each token in the source sentence, retrieves and outputs the most appropriate target tokens in the built-up phrase table.

In our approach, for the translation model input, we use both original order sentences and preordered sentences as the SMT input. Specifically, we replace the original SMT alignment method GIZA++[2] with SpanAlign and AWESoME-align, and follow the workflow of normal SMT.

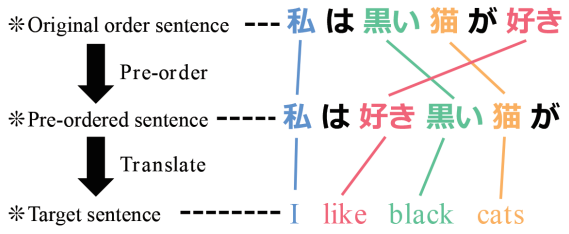


Figure 3: Transform the word order of the source Japanese to the target English before translation.

2 https://github.com/moses-smt/giza-pp

4 Experiments

4.1 Dataset

We use the ALT[15] (Asian Language Treebank)[3] as our main experiment corpus of setting Japanese on either the source side or target side, about 20K sentence pairs for each language pair. Others are Chinese, English, Bengali, Filipino, Hindi, Indonesian, Malay, and Thai. Parallel data are divided into the training data (18K), dev data (1K), and test data (1K).

4.2 Experimental Settings

4.2.1 Word Aligner Settings

For SpanAlign, we use the ALT Ja-En dev data of word alignment to fine-tune for Ja-En pair. For Ja-Zh, we use about 3,000 sentences of in-house word alignment data to fine-tune SpanAlign. Specific hyperparameters have followed as default, while the training batch size is set to 8 and the training epoch is set to 10. The average extraction threshold on bidirectional sides is 0.4. We did not conduct experiments based on other hyperparameters, so our choice may not be the most optimal.

For AWESoME-align, unsupervised data all comes from ALT dev data. The setting of the fine-tuning step of the training epoch as well as batch size is consistent with the SpanAlign.

Table 1: P, R, and F1 score of each alignment approaches of En-Ja. After fine-tuning, the precision of unsupervised AWESoME-align can reach the same level of supervised SpanAlign, while for the recall there remains still a gap.

Alignment Approach	P	R	F1 score
GIZA++	0.54	0.55	0.54
AWESoME-align	0.71	0.46	0.56
SpanAlign	**0.79**	**0.86**	**0.83**
AWESoME-align (fine-tuned)	**0.79**	0.58	0.67

4.2.2 Pointer Network Settings

Training data for the Pointer network are the

3 https://www2.nict.go.jp/astrec-att/member/mutiyama/ALT/

training data of original order sentences and pre-ordered sentences made by the alignments generated by SpanAlign. We use a 2-layer bidirectional LSTM, with a hidden state of 512 and an embedding state of 128. And we set the training batch size to 16, the learning rate to 3e-4, the training epoch is 10, max sequence length to 120. After training, the weighted Pointer Network is used to do the pre-ordering operation for test data sentences. We exploit RIBES[6], an efficient measure for automatically evaluating machine translation qualities based on the order of words, to evaluate the performance of the Pointer Network.

Table 2: RIBES result of Pointer Network trained by word alignments extracted by each approach, of transferring Japanese order into English order.

Model	RIBES
Manual Word Alignment	0.761
GIZA++	0.631
AWESoME-align	0.623
SpanAlign	**0.751**
AWESoME-align (fine-tuned)	0.722

4.3 Statistical Machine Translation

We use Moses[8][4] to make the phrase table, and the maximum length of each phrase is set to 3. We use a statistical-based trigram LM (Language Model), which is learned by target side sentences contained in the training part corpus, to ensure the fluency of the output sequence. Furthermore, we use dev data and MERT[13], joint with batch MIRA[2] to adjust the weight of the translation model. Note that all data used for SMT is token-based, we did not learn the BPE to further split the tokens.

4.4 Results

4.4.1 Pointer Network Performance

Because there is no ALT Chinese-Japanese manual alignment data that exists for evaluation, we only use Japanese and English data to verify the performance of the Pointer Network. Table 1 shows the F1 score between SpanAlign and AWESoME-align, demonstrating the high alignment accuracy.

Table 2 shows the result of the score of the preordered test data for transferring Japanese order into English order verified by RIBES. Here, we see ALT Japanese manual alignment data as the reference. From the results, Pointer Network trained with tokens extracted from SpanAlign and AWESoME-align (fine-tuned) are nearly the same as that of manual alignment, though the fine-tuned AWESoME-align is left behind. It can be considered that Pointer Network successfully learned certain language order features which are effective for the pre-ordering task.

Table 4: BLEU scores between the phrase-based translation of different phrase table length. 'Pre' represents for 'Pre-order input', while 'Ori' represents for 'Original input' and 'FT' is short for 'fine-tuned'.

Model		$L_{PT}=3$		$L_{PT}=5$	
		Pre	Ori	Pre	Ori
En → Ja	GIZA++	8.33	8.05	8.40	8.36
	SpanAlign	11.61	9.83	8.85	8.47
	AWESoME-align FT	10.86	9.15	12.35	9.88
	AWESoME-align FT + MERT	11.33	10.18	12.42	10.40
Zh → Ja	SpanAlign	10.11	8.24	8.59	7.84
	AWESoME-align FT	10.62	9.91	10.80	9.65
	AWESoME-align FT + MERT	10.80	10.20	11.24	9.67

4.4.2 Translation Accuracy

As a criterion to verify the translation accuracy, we use the BLEU[14] score. And we select Transformer-base and Transformer-small[17] as our baseline. Table 3 shows the translation accuracy of our proposed method with the alignment approach of SpanAlign and AWESoME-align (FT), also the accuracy of the baseline. From the two sets of the results, using

4 https://www.statmt.org/moses/

SpanAlign as an aligner is better than fine-tuned AWESoME-align for EnJa, nevertheless, using AWESoME-align as an aligner is better than Spanalign for Ja-Zh. The factor that causes this result is that, for En-Ja, both aligners are fine-tuned by dev data in ALT, while for Ja-Zh, the fine-tuning data of SpanAlign comes from in-house rather than ALT, so there are differences in specific domains. In addition, the results under various experimental conditions of our proposed framework are superior to Transformer-small.

4.4.3 Influence of the Length of Phrase Table

We explore the influence of the length of the phrase table on translation results in En-Ja and Zh-Ja directions, which is shown in Table 4. For SpanAlign, we are surprised to find that when a phrase table with length 5 is used for translation, the translation accuracy decreased compared with that of length 3. In contrast, when the length of the phrase table is changed from 3 to 5 using AWESOME-align in the En-Ja direction, the accuracy of the translation is improved regardless of whether pre-order or original order is used as the input. For the translation of AWESoME align in the Zh-Ja direction, the result with pre-order as input improved, while original order as input decreased

5 Conclusion and Future Work

In this paper, we propose a framework for low resource translation using SMT joint with highly accurate word alignment method SpanAlign and AWESoME-align rather than a sequence-to-sequence neural translation model. We use both

Table 3: BLEU score between two types of Transformer baseline and proposed method with alignment approach of SpanAlign and AWESoME-align (FT). FT represents for 'fine-tuned'. † for significant ($p < 0.05$) difference with baseline, and ‡ for significant ($p < 0.05$) difference of higher of the result of original input or pre-ordered input with the other side.

Model	Ja → En		En → Ja		Ja → Zh		Zh → Ja	
	Pre	Ori	Pre	Ori	Pre	Ori	Pre	Ori
Transformer-base	-	8.12	-	5.91	-	4.08	-	6.14
Transformer-small	-	7.02	-	10.64	-	6.33	-	9.83
PSMT+SpanAlign	8.74 †	9.23 † ‡	11.61 † ‡	9.83	7.17 †	8.36 † ‡	10.11 †	8.24
PSMT+AWESoME FT	8.36	9.22 † ‡	10.86 † ‡	9.15	9.30 †	9.49 †	10.62 † ‡	9.91
PSMT+AWESoME FT +MERT	8.37	9.65 † ‡	11.33 † ‡	10.18	9.29 †	10.04 † ‡	10.80 † ‡	10.20
Model	Ja → Bg		Bg → Ja		Ja → Fil		Fil → Ja	
	Pre	Ori	Pre	Ori	Pre	Ori	Pre	Ori
Transformer-base	-	6.03	-	4.45	-	3.26	-	4.65
Transformer-small	-	11.25	-	7.63	-	7.32	-	7.76
PSMT+AWESoME FT	10.97	10.64	8.86 †	9.28 †	7.84 † ‡	7.02	8.92 † ‡	6.75
PSMT+AWESoME FT +MERT	12.96 †	12.90 †	8.89 †	9.22 †	8.48 †	8.24 †	8.90 † ‡	7.85
Model	Ja → Hi		Hi → Ja		Ja → Id		Id → Ja	
	Pre	Ori	Pre	Ori	Pre	Ori	Pre	Ori
Transformer-base	-	5.93	-	4.92	-	2.04	-	5.68
Transformer-small	-	13.11	-	8.44	-	5.23	-	9.03
PSMT + AWESoME FT	13.20	13.33	11.32 † ‡	10.70 †	5.98 †	5.62	10.31 † ‡	7.19
PSMT+AWESoME FT +MERT	15.85 †	16.17 †	11.32 † ‡	10.84 †	6.83 †	6.71 †	10.35 † ‡	8.73
Model	Ja → Ms		Ms → Ja		Ja → Th		Th → Ja	
	Pre	Ori	Pre	Ori	Pre	Ori	Pre	Ori
Transformer-base	-	2.32	-	6.00	-	4.87	-	5.56
Transformer-small	-	5.18	-	9.28	-	7.25	-	7.84
PSMT + AWESoME FT	6.62 † ‡	6.07 †	9.93 † ‡	7.34	4.64	4.70	6.35	5.99
PSMT+AWESoME FT +MERT	6.54 †	7.54 †	9.86 † ‡	8.83	6.33	6.36	7.96 †	7.25

pre-ordered sentences which are preordered by Pointer Network, and original order sentences as input and perform the phrase-based translation. The results exceed the baseline of Transformer-base and Transformer-small except for Ja-Th and Th-Ja for high precision alignment. In future work, we will apply our approach with highly accurate word alignment to other language pairs.

References

[1] A. Bisazza and M. Federico. Surveys: A survey of word reordering in statistical machine translation: Computational models and language phenomena. *Computational Linguistics,* pp. 163–205, 2016.

[2] C. Cherry and G. Foster. Batch tuning strategies for statistical machine translation. In *Proc. 9th NAACL,* pp. 427–436, 2012.

[3] Z. Dou and G. Neubig. Word alignment by fine-tuning embeddings on parallel corpora. In *proc. 16th EACL,* pp. 2112– 2128, 2021.

[4] K. Duh, P. McNamee, M. Post, and B. Thompson. Benchmarking neural and statistical machine translation on low-resource African languages. In *Proc. 12th LREC,* pp. 2667–2675, 2020.

[5] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural Computation,* Vol. 9, No. 8, p. 1735–1780, 1997.

[6] H. Isozaki, T. Hirao, K. Duh, K. Sudoh, and H. Tsukada. Automatic evaluation of translation quality for distant language pairs. In *Proc. EMNLP,* pp. 944–952, 2010.

[7] Y. Kawara, C. Chu, and Y. Arase. Recursive neural network-based preordering for statistical machine translation and its analysis. *Journal of Natural Language Processing,* Vol. 26, No. 1, pp. 155–178, 2019.

[8] P. Koehn, H. Hoang, A. Birch, C. Callison-Burch, M. Federico, N. Bertoldi, B. Cowan, W. Shen, C. Moran, R. Zens, C. Dyer, O. Bojar, A. Constantin, and E. Herbst. Moses: Open source toolkit for statistical machine translation. In *Proc. 45th ACL Volume Proceedings of the Demo and Poster Sessions,* pp. 177–180, 2007.

[9] Z. Lin, X. Pan, M. Wang, X. Qiu, J. Feng, H. Zhou, and L. Li. Pre-training multilingual neural machine translation by leveraging alignment information. In *Proc. EMNLP,* pp. 2649–2663, 2020.

[10] Y. Liu, J. Gu, N. Goyal, X. Li, S. Edunov, M. Ghazvininejad, M. Lewis, and L. Zettlemoyer. Multilingual denoising pre-training for neural machine translation. *Transactions of the ACL,* pp. 726–742, 2020.

[11] R. Murthy, A. Kunchukuttan, and P. Bhattacharyya. Addressing word-order divergence in multilingual neural machine translation for extremely low resource languages. In *Proc. NAACL,* pp. 3868–3873, 2019.

[12] M. Nagata, K. Chousa, and M. Nishino. A supervised word alignment method based on cross-language span prediction using multilingual BERT. In *Proc. EMNLP,* pp. 555–565, 2020.

[13] F. Och. Minimum error rate training in statistical machine translation. In *Proc. 41st ACL,* pp. 160–167, 2003.

[14] K. Papineni, S. Roukos, T. Ward, and W. Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proc. 40th ACL,* pp. 311–318, 2002.

[15] H. Riza, M. Purwoadi, Gunarso, T. Uliniansyah, A. Ti, S. Aljunied, L. Mai, V. Thang, N. Thai, V. Chea, R. Sun, S. Sam, S. Seng, K. Soe, K. Nwet, M. Utiyama, and C. Ding. Introduction of the asian language

treebank. In *Proc. Oriental COCOSDA,* pp. 1-6, 2016.

[16] S. Sen, K. Gupta, A. Ekbal, and P. Bhattacharyya. Multilingual unsupervised NMT using shared encoder and languagespecific decoders. In *Proc. 57th ACL,* pp. 3083-3089, 2019.

[17] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. Gomez, U. Kaiser, and I. Polosukhin. Attention is all you need. In *Proc. 30th NIPS,* pp. 5998-6008, 2017.

[18] O. Vinyals, M. Fortunato, and N. Jaitly. Pointer networks. In *Proc. 28th NIPS,* pp.1-9, 2015.

[19] L. Xue, N. Constant, A. Roberts, M. Kale, R. Al-Rfou, A. Siddhant, A. Barua, and C. Raffel. mT5: A massively multilingual pre-trained text-to-text transformer. In *Proc. 15th NAACL,* pp. 483-498, 2021.

[20] P. Zaremoodi and G. Haffari. Adaptively scheduled multitask learning: The case of low-resource neural machine translation. In *Proc. 3rd WNGT,* pp. 177-186, 2019.

高精度の単語アライメントを利用した SMT ベース低資源機械翻訳（抄録）

近年、従来の統計的機械翻訳（Statistical Machine Translation；SMT）に代わって，ニューラルネットワーク機械翻訳（Neural Machine Translation；NMT）モデルが盛んに研究されている．通常、NMT においては、大規模な訓練用資源が不可欠であるため、低資源言語翻訳においては、高い性能を出すことが難しい。NMT におけるこれまでの低資源言語翻訳に関する研究としては、主に以下のアプローチの研究が行われてきた。一つのアプローチとして、大規模な単言語コーパスを用いて単語埋め込み表現を学習し、その後、小規模な二言語対訳コーパスを用いて、下流タスクにおける fine-tuning を行うものがある。一方、教師なしのアプローチでは、原言語と目的言語の間で単語埋め込みの対応がとれるように共通の多言語埋め込み空間に対応させ、二言語間で類似の意味を持つ語の埋め込みが近くなるようにするとともに、反復的に疑似の原言語文と目的言語文を生成し、訓練の過程を行う。しかし、これらの方式には多くの計算量が必要な点が問題である。

本論文では、低資源言語翻訳方式として、SMT、および、高精度単語アライメント手法である SpanAlign または AWESoME-align に基づく枠組みを提案する。この枠組みにおいては、NMT モデルではなく、フレーズベースの SMT モデルを用いる。本論文では特に，高精度単語アライメント手法である SpanAlign または AWESoME-align によって得られた単語アライメントを入力として SMT モデルを訓練することにより、高精度な SMT モデルを実現している。さらに、入力文の語順を目的言語文の語順に変換するためのニューラルネットワークとして Pointer Network を用いる。各言語対において、原言語文の語順のままで SMT モデルを適用した場合、および、目的言語文の語順へ変換後 SMT モデルを適用した場合の二通りの評価を行った。SMT モデルの訓練用コーパスとして、Asian Language Treebank コーパスの対訳文 2 万文を用いた。言語対として、日本語を原言語または目的言語として、中国語、英語、ベンガル語、フィリピン語、ヒンディー語、インドネシア語、マレー語、タイ語との間の翻訳において評価を行った。評価結果においては、提案手法は、全言語対において Transformer-base を上回り、言語対 Ja-Th を除いては、Transformer-small を上回った。