Learning of Linear Ordering Problems and its Application to J-E Patent Translation in NTCIR-9 PatentMT

Shuhei Kondo
Nara Institute of Science and Technology
8916-5 Takayama, Ikoma, Nara 630-0192, Japan
shuhei-k@is.naist.jp

Mamoru Komachi
Nara Institute of Science and Technology
8916-5 Takayama, Ikoma, Nara 630-0192, Japan
komachi@is.naist.jp

Yuji Matsumoto
Nara Institute of Science and Technology
8916-5 Takayama, Ikoma, Nara 630-0192, Japan
matsu@is.naist.jp

Katsuhito Sudoh
NTT Communication Science Laboratories
2-4 Hikaridai, Seika-cho, Soraku-gun, Kyoto 619-0237 Japan
sudoh.katsuhito@lab.ntt.co.jp

Kevin Duh
NTT Communication Science Laboratories
2-4 Hikaridai, Seika-cho, Soraku-gun, Kyoto 619-0237 Japan
kevin.duh@lab.ntt.co.jp

Hajime Tsukada
NTT Communication Science Laboratories
2-4 Hikaridai, Seika-cho, Soraku-gun, Kyoto 619-0237 Japan
tsukada.hajime@lab.ntt.co.jp

ABSTRACT

This paper describes the patent translation system submitted for the NTCIR-9 PatentMT task. We applied the Linear Ordering Problem (LOP) based reordering model [16] to Japanese-to-English translation to deal with the substantial difference in the word order between the two languages.

Categories and Subject Descriptors

I.2.7 [**Artificial Intelligence**]: Natural Language Processing—*machine translation*

General Terms

Experimentation

Keywords

Statistical Machine Translation, Reordering, Preprocessing.

Team Name

[NAIST]

Subtasks/Languages

[Japanese to English]

External Resources Used

[Moses][GIZA++][SRILM][MeCab]

1. INTRODUCTION

While phrase-based statistical machine translation can model short-distance reordering within the phrase fairly well, long-distance reordering remains one of the major problems in SMT, especially for translations between language pairs with substantially different word order.

Japanese and English are an example of such a pair, the former being an SOV language and the latter an SVO language. Translations between such language pairs are difficult because the decoder often misses long-distance reordering required for an appropriate translation, partly due to the distortion limit that is commonly adopted by phrase-based decoders to reduce the search space.

Reordering by preprocessing is one of the common approaches to deal with long-distance reordering required for translation between languages with different word order. Works in this direction can be classified into syntax-based [3, 9], chunk-based [1, 18] and word-based [9, 16] approaches. The syntax-based and the chunk-based approaches depend on syntactic parsing and base-phrase chunking, and therefore they usually require rich language resources such as treebanks. In contrast, word-based approaches only use part-of-speech (POS) tagger or simple rules, and thus it is applicable to languages and domains with limited resources.

We applied one of the word-based approaches, the LOP-based word-level reordering model [16], for Japanese-to-English translation in the system submitted to the NTCIR-9 patent machine translation task [7].

2. LINEAR ORDERING PROBLEM BASED REORDERING

Tromble and Eisner (2009) [16] proposed a word-level reordering model based on Linear Ordering Problem (LOP) and an efficient decoding algorithm called "local search".

2.1 The LOP-based Model

Given an input sentence $\boldsymbol{w} = w_1 w_2 ... w_n$, a permutation $\boldsymbol{\pi} = \pi_1 \pi_2 ... \pi_n$ is any reordering of the tokens in $\boldsymbol{w}$. The LOP-based model is defined so that it assigns a high score to the permutation $\boldsymbol{\pi}$ that corresponds well to the word order of the target language. To construct such model, a pairwise preference matrix $B_{\boldsymbol{w}}$ is constructed for each sentence $\boldsymbol{w}$, whose entries are

$$B_{\boldsymbol{w}}[l, r] \stackrel{def}{=} \boldsymbol{\theta} \cdot \boldsymbol{\phi}(\boldsymbol{w}, l, r) \quad (1)$$

where $\boldsymbol{\theta}$ is a vector of weights, $\boldsymbol{\phi}$ is a vector of feature functions considering the input sentence $\boldsymbol{w}$ and its functions such as POS tags, l and r are the positions of left and right

Table 1: Feature templates for $B_w[l,r]$ (w_l is the lth word and t_l is its POS tag. b matches any index with $l < b < r$.) Each combination is conjoined with the distance between the words, $r-l$. Distances are binned into 1, 2, 3, 4, 5, 5 $< d \leq$ 10 and 10 $< d$.

t_{l-1}	w_l	t_l	t_{l+1}	t_b	t_{r-1}	w_r	t_r	t_{r+1}
	×	×				×	×	
	×	×				×		
	×					×	×	
	×	×					×	
		×				×	×	
	×					×		
		×					×	
	×	×						
						×	×	
	×							
						×		
		×						
							×	
		×		×			×	
		×	×		×		×	
		×	×				×	
		×	×				×	×
		×					×	×
×		×					×	×
×		×					×	
×		×				×	×	
		×				×	×	

word in the input sentence. $B_{\boldsymbol{w}}[l,r] > B_{\boldsymbol{w}}[r,l]$ means that the model prefers a permutation in which the order of the l-th word and r-th word is the same as the input, while $B_{\boldsymbol{w}}[l,r] < B_{\boldsymbol{w}}[r,l]$ means that the order of l-th word and r-th word should be reversed. The score for a permutation $\boldsymbol{\pi}$ is defined as

$$\text{score}(\boldsymbol{\pi}) \stackrel{def}{=} \sum_{i,j:1 \leq i < j \leq n} B_{\boldsymbol{w}}[\pi_i, \pi_j], \quad (2)$$

where $B_{\boldsymbol{w}}[r,l]$ can be considered 0 for any $l < r$, because subtracting $B_{\boldsymbol{w}}[r,l]$ from both $B_{\boldsymbol{w}}[l,r]$ and $B_{\boldsymbol{w}}[r,l]$ will reduce the scores of all permutations by the same amount.

To calculate the score for (2), the weight vector $\boldsymbol{\theta}$ in (1) is learned discriminatively with the reference reordering obtained from automatic word alignments. Each source token is assigned an integer key that corresponds to the position of the leftmost target token which is aligned to it, or 0 if it is not aligned to any target token. The order of l-th token and the r-th token is swapped in the reference reordering if the key for the former is larger than the latter. Table 1 shows the feature templates for $\boldsymbol{\phi}$ used by [16], which is adapted from [11].

2.2 Local Search

An algorithm called "local search" is used to reorder the source sentence according to the LOP-based model. Given an input sentence $\boldsymbol{w}$, for each span $i...k$ which yields the substring $w_{i+1}w_{i+2}...w_k$, the score $\max(0, \Delta_{i,j,k})$ for its subspan $i...j$, $j...k$ is calculated according to the model for all pairs of subspans. $\Delta_{i,j,k}$ is the score of swapping the subspans with the base score $\Delta_{i,j,k} = 0$ for $i = j$ and $j = k$, and a positive score means that they should be swapped. Starting with the spans that consist of one token and recursively expanding the spans, local search finds the best permutation of the input sentence under the ITG [17] constraints according to the LOP-based model in $O(n^3)$ time, where n is the length of the input sentence.

Table 2: BLEU scores on the development set with varying distortion limits.

Word Order	Distortion Limit	BLEU
LOP-based	6	29.52
LOP-based	10	29.71
LOP-based	20	29.56
Original	10	28.56
Original	20	29.44
Original	30	29.30

Table 3: Average execution time (in seconds) of the LOP-based reordering of 10,000 sentences in 10 trials. (a) is for the first 10,000 sentences in the training set, whose average length is 19.60 words, and (b) is for 10,000 sentences from the 1,000,001st to 1,010,000th sentence, whose average length is 36.29 words.

	(a)	(b)
time(sec)	56.53	337.23

Table 4: Official automatic evaluation results

System	BLEU	NIST	RIBES
LOP-based	27.82	7.4348	0.730743
BASELINE1	28.95	7.7696	0.70644
BASELINE2	28.61	7.7562	0.675831

Table 5: Official human evaluation results

System	Adequacy	Acceptability
LOP-based	2.610	0.472
BASELINE1	2.617	0.474
BASELINE2	2.427	0.447

Table 6: BLEU scores on the NTCIR-9 PatentMT test set

Word Order	BLEU	NIST	RIBES
LOP-based	27.82	7.4348	0.730737
Original	27.37	7.5648	0.675202

3. EXPERIMENTS

3.1 Data

We used the training set of NTCIR-7 Patent Translation Task (1.8 million sentence pairs) [5, 6] to train the translation model, and its English part for the language model; data from 2001 to 2005 were not used due to time limitation. Sentences with more than 80 tokens were also excluded. We corrected some obvious errors in its Japanese part, such as prolonged sound marks replaced by minus signs, using regular expressions.

To learn the LOP-based reordering model, we picked 300,000 sentence pairs with high sentence alignment score from the training set. The weight vector $\boldsymbol{\theta}$ was learned using Averaged Perceptron [2] with the 300,000 pairs. We limited the amount of corpus here because automatic word alignments obtained from sentence pairs with low alignment score are

likely to be less reliable and they can be harmful to the reordering model.

Then training, development and test sentences were reordered by local search with the learned weights, and the translation model was retrained on the reordered training set. Table 2 shows the BLEU [14] scores of the LOP-based system on the development set with varying distortion limits, compared with the system with original word order and same amount of corpus. The result shows that the LOP-based system requires smaller distortion limit. As the LOP-based system performed best with $d = 10$ on the development set, we used this value for the test.

Table 3 shows the average execution time required for reordering of source sentences. We can see reordering by preprocessing in our system can be performed quite efficiently, even with the naive implementation written in Ruby for reordering.

3.2 Tools

Japanese sentences were tokenized and POS-tagged using MeCab version 0.98[1]. We used mecab-ipadic-2.7.0-20070801 as the dictionary, with a slight modification on costs for the words whose POS tag is exclamation, to penalize the words that are unlikely to appear in patent texts.

We used GIZA++ (version 1.0.5) [13] for word alignment, SRILM (version 1.5.11) [15] for construction of 5-gram language model and Moses (revision 3947) [10] for training of translation models and decoding. Parameters for each model were tuned using minimum error-rate training [12] with BLEU as the objective. Recaser was trained in similar manner to organizer's baseline with the same amount of data as the training set of our system.

3.3 Results

Table 4 shows the official scores of our system based on automatic evaluation metrics, compared with baseline systems provided by the organizer. Our system achieved lower scores in BLEU and NIST [4] metric, and higher score in RIBES [8] metric than both baseline systems.

Table 5 shows the official scores of our system based on human evaluations. Our system's performance was comparable to BASELINE1 (hierarchical phrase based) and slightly better than BASELINE2 (phrase-based). This result seems encouraging, because our system was trained on less resources and its procedure after preprocessing was quite similar to that of BASELINE2.

4. ANALYSIS

We compared the LOP-based system with a system trained with the same amount of corpus that is in the original word order, because the LOP-based system was trained on smaller corpus compared to the BASELINE systems provided by the organizer as mentioned above in section 3.

Table 6 shows the BLEU, NIST and RIBES scores of both systems. While the reordered system's performance was worse in the NIST score, it performed better in BLEU and RIBES scores. Both in this result and the result in the official test, the LOP-based model seems to perform well in RIBES score and poorly in NIST score.

We looked at example outputs of both systems to analyze the reason for this tendency. Table 7 shows several examples of original source sentences, reordered inputs, outputs of the system trained with original word order, and outputs of the LOP-based system with references, in which the LOP-based reordering model worked well. In these examples, the reordered inputs can be translated almost monotonically.

However, we can also see that some case-markers are moved out of their context in the reordered inputs. Moreover, in the last example, tokens forming the Japanese compound word for "engine control system" are separated as a result of reordering. Though the effects on overall reordering were small in these examples, such displacement can harm the quality of translation.

The gap between BLEU and NIST scores can possibly be attributed to this displacement of case-markers and compound words, because NIST metric gives more weight to rare n-grams and therefore is more sensitive than BLEU metric to such displacement, especially in the cases with compound words. One way to deal with this problem might be to change the unit of reordering, for example from words to chunks. Extending the unit of reordering to ones containing more than one words might contribute to reduction of the time required for reordering, as local search is $O(n^3)$ algorithm and depends on the number of units to be reordered.

High RIBES score may indicate that overall reordering performed by our system is reasonably good, because RIBES metric is designed so as to evaluate the global word reordering well, which is required for translation with distant language pairs. Parameter tuning with RIBES score instead of BLEU score as the objective might further improve the global reordering.

5. CONCLUSION

In this paper, we applied the LOP-based word-level reordering for Japanese-to-English translation. In automatic evaluation, our system performed worse in two metrics and better in one metric compared to both baseline systems. When compared with the same amount of training data, our system performed better in two metrics than the system with original word order. In human evaluation, our system performed better than phrase-based baseline and similarly to hierarchical phrase-based one, despite the fact that our system was trained on less amount of data.

Besides the problem of global reordering, displacement of case-markers and tokens that form compound words was observed in our system. We think one way of the future work will be to find the model that can prevent such displacement.

6. ACKNOWLEDGMENTS

The authors wolud like to thank Katsuhiko Hayashi for valuable comments and discussions, and the anonymous reviewer for insightful comments and suggestions.

7. REFERENCES

[1] A. Bisazza and M. Federico. Chunk-based verb reordering in VSO sentences for Arabic-English statistical machine translation. In *Proceedings of the Joint Fifth Workshop on Statistical Machine Translation and MetricsMATR*, pages 235–243, 2010.

[2] M. Collins. Discriminative training methods for hidden markov models: Theory and experiments with perceptron algorithms. In *Proceedings of the 2002*

[1] http://mecab.sourceforge.net/

Table 7: Sample of reordered inputs and outputs of our system

Original source sentence	そして 、 スピン バルブ 型 巨大 磁気 抵抗 効果 素子 対 2 ， 3 ， 4 ， 5 それぞれ の 他方 の 端 部 間 に 測定 用 電圧 Vcc が 印加 さ れ て いる 。
Reordered input	そして 、 の に て が 印加 さ れ いる 測定 用 電圧 Vcc 間 他方 端 部 の それぞれ スピン バルブ 型 巨大 磁気 抵抗 効果 素子 対 2 ， 3 ， 4 ， 5 。
Original output	Then, the spin valve type giant magnetoresistance effect element pair 2, 3, 4, 5, respectively, between the other end of the measuring voltage Vcc is applied.
LOP-based output	Then, the measuring voltage Vcc is applied to between the other end of the spin-valve type giant magnetoresistive element pair 2, 3, 4, 5.
Reference	A voltage Vcc for measurement is applied between the other ends of the spin-valve giant magneto-resistive element pairs 2, 3, 4 and 5.
Original source sentence	横 方向 スキャン 装置 70 の 動作 は 、 レーダ 装置 10 の 制御 演算 部 11 にて 制御 さ れる 。
Reordered input	動作 の 横 方向 スキャン 装置 70 は 制御 さ れる にて の 制御 演算 部 11 、 レーダ 装置 10 。
Original output	The operation of the radar apparatus 10, the control arithmetic unit 11 in the lateral direction is controlled by a scanning device 70.
LOP-based output	The operation of the lateral direction scan unit 70 is controlled by the control arithmetic unit 11 of the radar apparatus 10.
Reference	The operations of the horizontal-direction scanning device 70 are controlled by the control and signal processing unit 11 of the radar apparatus 10.
Original source sentence	この 第 3 過程 における 位相 変調 は 、集 光 レンズ 等 から なる 第 2 の 光学 素子 12 を 用い て 実現 できる 。
Reordered input	この 位相 変調 における 第 3 過程 は できる 実現 を 用い て の 第 2 光学 素子 12 から なる 、 集 光 レンズ 等 。
Original output	In the third process, a condenser lens or the like using the phase modulation of the second optical element 12 can be realized.
LOP-based output	This phase modulation in the third process is implemented using a second optical element 12 comprising the condensing lens or the like.
Reference	The phase modulation in this third process can be realized using a second optical element 12 composed of a focusing lens or the like.
Original source sentence	マップ データ C は 、 センサ 故障 など エンジン 制御 システム に 異常 が 発生 し た 場合 に 使用 する 特性 変換 係数 用 の マップ データ で ある 。
Reordered input	マップ データ C は で の マップ データ 用 ある 特性 変換 する 係数 使用 に 場合 異常 が 発生 し たに 、 エンジン など センサ 故障 制御 システム 。
Original output	Map data C, the sensor failure or abnormality has occurred in the engine control system used in a case where a characteristics conversion coefficient map data.
LOP-based output	The map data C is a map data for the characteristic conversion coefficient used in a case where abnormality is generated in the engine sensor failure control system.
Reference	The map data C is the map data for the characteristic conversion coefficient used when a defect occurs in the engine control system, such as a failure of a sensor.

Conference on Empirical Methods in Natural Language Processing, pages 1–8, 2002.

[3] M. Collins, P. Koehn, and I. Kučerová. Clause restructuring for statistical machine translation. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, pages 531–540, 2005.

[4] G. Doddington. Automatic evaluation of machine translation quality using n-gram co-occurrence statistics. In *Proceedings of the second international conference on Human Language Technology Research*, pages 138–145, 2002.

[5] A. Fujii, M. Utiyama, M. Yamamoto, and T. Utsuro. Overview of the Patent Translation Task at the NTCIR-7 Workshop. In *Proceedings of NTCIR-7 Workshop Meeting*, pages 389–400, 2008.

[6] A. Fujii, M. Utiyama, M. Yamamoto, T. Utsuro, T. Ehara, H. Echizen-ya, and S. Shimohata. Overview of the Patent Translation Task at the NTCIR-8 Workshop. In *Proceedings of NTCIR-8 Workshop Meeting*, pages 371–376, 2010.

[7] I. Goto, B. Lu, K. P. Chow, E. Sumita, and B. K. Tsou. Overview of the Patent Machine Translation Task at the NTCIR-9 Workshop. In *Proceedings of NTCIR-9 Workshop Meeting*, 2011.

[8] H. Isozaki, T. Hirao, K. Duh, K. Sudoh, and H. Tsukada. Automatic Evaluation of Translation Quality for Distant Language Pairs. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 944–952, 2010.

[9] J. Katz-Brown and M. Collins. Syntactic reordering in preprocessing for Japanese → English translation: MIT system description for NTCIR-7 patent translation task. In *Proceedings of NTCIR-7 Workshop Meeting*, pages 409–414, 2008.

[10] 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 *Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster*

and Demonstration Sessions, pages 177–180, 2007.

[11] R. McDonald, K. Crammer, and F. Pereira. Spanning Tree Methods for Discriminative Training of Dependency Parsers. Technical Report MS-CIS-05-11, UPenn CIS, 2005.

[12] F. J. Och. Minimum error rate training in statistical machine translation. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics*, volume 1, pages 160–167, 2003.

[13] F. J. Och and H. Ney. A systematic comparison of various statistical alignment models. *Computational linguistics*, 29(1):19–51, 2003.

[14] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu. Bleu: a Method for Automatic Evaluation of Machine Translation. In *Proceedings of 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, 2002.

[15] A. Stolcke. SRILM-an extensible language modeling toolkit. In *Proceedings of the International Conference on Spoken Language Processing*, volume 2, pages 901–904, 2002.

[16] R. Tromble and J. Eisner. Learning Linear Ordering Problems for Better Translation. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pages 1007–1016, 2009.

[17] D. Wu. Stochastic inversion transduction grammars and bilingual parsing of parallel corpora. *Computational Linguistics*, 23(3):377–403, 1997.

[18] Y. Zhang, R. Zens, and H. Ney. Improved chunk-level reordering for statistical machine translation. In *Proceedings of the International Workshop on Spoken Language Translation*, pages 21–28, 2007.