Japanese Predicate Conjugation for Neural Machine Translation

Michiki Kurosawa, Yukio Matsumura, Hayahide Yamagishi and Mamoru Komachi
Tokyo Metropolitan University
Hino city, Tokyo, Japan
{*kurosawa-michiki, matsumura-yukio, yamagishi-hayahide*}@ed.tmu.ac.jp
komachi@tmu.ac.jp

Abstract

Neural machine translation (NMT) has a drawback in that can generate only high-frequency words owing to the computational costs of the softmax function in the output layer.

In Japanese-English NMT, Japanese predicate conjugation causes an increase in vocabulary size. For example, one verb can have as many as 19 surface varieties. In this research, we focus on predicate conjugation for compressing the vocabulary size in Japanese. The vocabulary list is filled with the various forms of verbs. We propose methods using predicate conjugation information without discarding linguistic information. The proposed methods can generate low-frequency words and deal with unknown words. Two methods were considered to introduce conjugation information: the first considers it as a token (**conjugation token**) and the second considers it as an embedded vector (**conjugation feature**).

The results using these methods demonstrate that the vocabulary size can be compressed by approximately 86.1% (Tanaka corpus) and the NMT models can output the words not in the training data set. Furthermore, BLEU scores improved by 0.91 points in Japanese-to-English translation, and 0.32 points in English-to-Japanese translation with ASPEC.

1 Introduction

Neural machine translation (NMT) is gaining significant attention in machine translation research because it produces high-quality translation (Bahdanau et al., 2015; Luong et al., 2015a). However, because NMT requires massive computational time to select output words, it is necessary to reduce the vocabulary in practice by using only high-frequency words in the training corpus. Therefore, NMT treated not only **unknown words**, which do not exist in the training corpus, but also **OOV**, which can not consider words to NMT's computational ability, as unknown word token[1].

Two approaches were proposed to address this problem: backoff dictionary (Luong et al., 2015b) and byte pair encoding, or BPE (Sennrich et al., 2016). However, because the backoff dictionary is a post-processing method to replace OOV, it is not a fundamental solution. BPE can eliminate unknown words by dividing a word into partial strings; however, there is a possibility of loss of linguistic information such as loss of the meaning of words.

In Japanese grammar, the surfaces of verb, adjective, and auxiliary verb change into different forms by the neighboring words. This phenomenon is called "conjugation," and 18 conjugation patterns can be formed at maximum for each word. We consider the conjugation forms as the vocabulary of NMT using Japanese language because the Japanese morphological analyzer divides a sentence into words based on conjugation forms. The vocabulary set in the NMT model must have all conjugation forms for generating fluent sentences.

In this research, we propose two methods using predicate conjugation information without discarding linguistic information. These methods can not only reduce OOV words, but also deal with unknown words. In addition, we consider a method to introduce part-of-speech (POS) information other than predicate. We found this method is related to source head information.

The main contributions of this paper are as follows:

[1] In this paper, we denote a word not appearing in the training corpus as "unknown word," and a word treated as an unknown low-frequency word as "OOV."

Proceedings of NAACL-HLT 2018: Student Research Workshop, pages 100–105
New Orleans, Louisiana, June 2 - 4, 2018. ©2017 Association for Computational Linguistics

語幹	未然形	連用形	終止形	連体形	仮定形	命令形
Stem	Irrealis	Continuative	Terminal	Attributive	Hypothetical	Imperative
走る (*hashi-ru*; run)	走ら (*hashi-ra*) 走ろ (*hashi-ro*)	走り (*hashi-ri*)	走る (*hashi-ru*)	走る (*hashi-ru*)	走れ (*hashi-re*)	走れ (*hashi-re*)
歩く (*aru-ku*; walk)	歩か (*aru-ka*) 歩こ (*aru-ko*)	歩き (*aru-ki*)	歩く (*aru-ku*)	歩く (*aru-ku*)	歩け (*aru-ke*)	歩け (*aru-ke*)
する (*su-ru*; do)	せ (*se*) し (*shi*)	し (*shi*)	する (*su-ru*)	する (*su-ru*)	すれ (*su-re*)	しろ (*shi-ro*) せよ (*se-yo*)

Table 1: Leverage table of verb.

- The proposed NMT reduced the vocabulary size and improved BLEU scores particularly in small- and medium-sized corpora.
- We found that conjugation features are best exploited as tokens rather than embeddings and suggested the connection between the position of the token and linguistic properties.

2 Related work

Backoff dictionary. Luong et al. (2015b) proposed a method of rewriting an unknown word token in the output into an appropriate word using a dictionary. This method determines a corresponding word using alignment information between an output sentence and an input sentence and rewrites the unknown word token in the output using the dictionary. Therefore, it does not allow NMT to consider the meaning of OOVs. However, this method can be used together with the proposed method, which results in the further reduction of unknown words.

Byte pair encoding. Sennrich et al. (2016) proposed a method to construct vocabulary by splitting all the words into characters and re-combining them based on their frequencies to make subword unit. Because all words can be split into known words based on characters, this method has an advantage in that OOV words disappear. However, because coupling of subwords depends on frequency, grammatical and semantic information is not taken into consideration. Incidentally, Japanese has many characters especially kanji; therefore, there might exist unknown characters that do not exist in the training corpus even after applying BPE.

Input feature. Sennrich and Haddow (2016) proposed a method to add POS information and dependency structure as embeddings with the aim of explicitly learning syntax information in NMT. However, it can only be applied to the input side.

3 Japanese predicate conjugation

Japanese predicates consist of stems and conjugation suffixes. In the vocabulary set obtained by conventional word segmentation, they are treated as different words. Therefore, the vocabulary set is occupied with predicates which have similar meaning but different conjugation.

As an example, a three-type conjugation table is shown in Table 1. In this way, conjugation represents many expressions with only a subtle difference in meaning. Due to the Japanese writing system, most of the predicates do not share conjugation suffixes even though they share the same conjugation patterns. Comparing "走る (run)" and "歩く (walk)", if one wants to share the conjugation suffixes using BPE, it is necessary to represent these words using Latin alphabets instead of phonetic characters, or kana. In addition, a special verb "する (do)" cannot share the conjugation suffixes with these words even using BPE. Therefore, we cannot divide the predicates into the stems and shared conjugation suffixes using BPE.

In the proposed method, we handle them collectively. Since types of conjugation are limited, we can deal with every types. All conjugation forms can be consolidated into one lemma, and OOV can be reduced[2]. Furthermore, by treating a lemma and conjugation forms as independent words, it is possible to represent the predicates which we were observed a few times on the training corpus by combining lemmas and conjugation forms found in the training corpus.

In this research, MeCab[3] is used as a Japanese morphological analyzer, and the morpheme information adopts the standard of IPADic. Specifically, "*surface form*", "*POS (coarse-grained)*", "*POS (fine-grained)*", "*conjugation type*", "*conju-*

[2] Derivational grammar (Ogawa et al., 1998) to unify multiple conjugation forms, but it cannot distinguish between plain and attributive forms and imperfective and continuative forms if they have the same surface.

[3] `https://github.com/taku910/mecab`

gation form”, and “*lemma*” are used. Hereafter, predicates represent verbs, adjectives, and auxiliary verbs.

4 Introducing Japanese predicate conjugation for NMT

We propose two methods to introduce conjugation information: in the first method, it is treated as a token (**conjugation token**) and in the second, it is treated as concatenation of embeddings (**conjugation feature**). Moreover we considere to introduce POS information into all words (**POS token**).

4.1 Conjugation token

In this method, lemmas and conjugation forms are treated as tokens. A conjugation form is introduced as a special token with which its POS can be distinguished from other tokens.

In this method, the special token also occupies a part of the vocabulary. However, as there are only 55 tokens[4] at maximum in the IPADic standard, the influence is negligible compared to the vocabulary size that can be reduced. Moreover, because the stem and its conjugation suffix are explicitly retrieved, the output can be restored at any time. For example, these are converted as follows.

走る	→	走る (run)	<動詞・基本形> (verb–plain)
走れ	→	走る (run)	<動詞・命令形> (verb–imperative)
だ	→	だ (COPULA)	<助動詞・体言接続> (aux.verb–attributive)

4.2 Conjugation feature

In this method, we use a conjugation form as a feature of input side. Specifically, “*POS (coarse-grained)*”, “*POS (fine-grained)*”, and “*conjugation forms*” are used in addition to the lemma. Moreover, this information is added to words other than predicates. These features are first represented as one-hot vectors, and the learned embedding vectors are concatenated and used.

This method has an advantage in that it does not waste vocabulary size; however, because it is not trivial to restore a word from embeddings, it can be adopted to the source side only.

4.3 POS token

As a natural extension to Conjugation token, we introduce POS information into all words in addition to conjugation information. We use POS information and conjugation information in the same manner to Conjugation token. We propose three methods to incorporate POS information as special tokens.

Corpus	train	dev	test	Max length
NTCIR	1,638,742	2,741	2,300	60
ASPEC	827,503	1,790	1,812	40
Tanaka	50,000	500	500	16

Table 2: Details of each corpus.

Suffix token. This method introduces POS and conjugation information behind each word as a token.

Prefix token. This method introduces POS and conjugation information in front of each word as a token.

Circumfix token. This method introduces POS information in front of each word and conjugation information behind each word as a token.

Example sentences are shown below:

Baseline

私 は 走る 。(I run .)

Suffix token

私 <noun> は <particle> 走る <verb-plain> <verb> 。 <symbol>

Prefix token

<noun> 私 <particle> は <verb> <verb-plain> 走る <symbol> 。

Circumfix token

<noun> 私 <particle> は <verb> 走る <verb-plain> <symbol> 。

5 Experiment

We experimented two baseline methods (with and without BPE) and two proposed methods. Each experiment was conducted four times with different initializations. We report the average performance over all experiments.

We used three data sets: NTCIR PatentMT Parallel Corpus - 10 (Goto et al., 2013), Asian Scientific Paper Excerpt Corpus (Nakazawa et al., 2016), and Tanaka Corpus (Excerpt, Preprocessed)[5]. The details of each corpus are shown in Table 2. Only in Tanaka, English sentences were

[4] Verb: 18, Adjective: 14, Auxiliary verb: 22

[5] http://github.com/odashi/small_parallel_enja

Method		Japanese - English			English - Japanese		
		NTCIR	ASPEC	Tanaka	NTCIR	ASPEC	Tanaka
Baseline	w/o BPE	33.87	20.98	30.23	36.41	29.57	30.25
	BPE only Japanese	**34.17**	21.10	30.43	35.96	28.96	28.66
	BPE both sides	-	21.43	30.45	-	30.93	29.27
	BPE only English	-	20.55	30.13	-	30.59	29.15
Only predicate conjugation information	(4.1) Conjugation token	33.96	21.47	32.47	**36.48**	**29.89**	30.46
	(4.2) Conjugation feature	33.84	21.33	30.35	N/A	N/A	N/A
Using predicate conjugation information and all POS information	(4.3) Suffix token	-	21.49	31.82	-	29.77	**31.47**
	(4.3) Prefix token	-	21.61	32.16	-	29.02	30.36
	(4.3) Circumfix token	-	**21.89**	**32.96**	-	28.89	31.07

Table 3: BLEU scores of each experiment (average of four runs). The best score in each corpus is made bold (expect for BPE "both" and "only English").

already lowercased; hence, truecase was not used. As for ASPEC, we used only the first one million sentences sorted by sentence alignment confidence. Japanese sentences were tokenized by the morphological analyzer MeCab (IPADic), and English sentences were preprocessed by Moses[6] (tokenizer, truecaser). As for the training corpus, we deleted sentences that exceeded the maximum number of tokens each sentence shown in Table 2.

We used our implementation[7] based on Luong et al. (2015a) as the baseline. Hyperparameters are as follows. If the setting differs in the corpus, it is written in the order of NTCIR / ASPEC / Tanaka.

Optimization: AdaGrad, Learning rate: 0.01,
Embed size: 512, Hidden size: 1,024,
Batch size: 128, Maximum epoch: 15 / 15 / 30,
Vocab size: 30,000 / 30,000 / 5,000,
Output limit: 100 / 100 / 40

The setting of each experiment except the baseline is shown below. We used the same setting as the baseline unless otherwise specified.

Byte pair encoding. We conducted an experiment using BPE as the comparative method. BPE was applied to the Japanese side only for making a fair comparison with the proposed method.

The number of merge operations in both NTCIR and ASPEC was set to 16,000 and in Tanaka, the number was set to 2,000. As a result, OOV did not exist in all corpora because the size of Japanese vocabulary is smaller than that of BPE.

Conjugation token. Because the output of English–Japanese translation includes special tokens, we evaluate it by restoring the results with rules using IPADic. The restoration accuracy is 100%. If the output has only a lemma, it is converted into the plain form, and if it has a conjugation token only, the token is deleted from the output.

Conjugation feature. Because this method can solely be adopted to the source side, only Japanese-to-English translation was performed. To restrict the embed size to 512, the size of each feature was set to POS (coarse-grained): 4, POS (fine-grained): 8, conjugation form: 8, lemma: 492.

POS token. We increased the output limit by 2.5 times in English-to-Japanese translation because of additional POS tokens attached to all words.

We used the same restoration rules as for Conjugation token to treat special tokens.

We evaluated POS features in only ASPEC and Tanaka owing to time constraints.

6 Discussion

6.1 Translation quality

The results of BLEU score (Papineni et al., 2002) are shown in Table 3. Compared to the baseline without BPE, Conjugation token improved in BLEU score on all corpora and in both translation directions. In addition, Conjugation token outperformed the baselines with BPE with an exception on NTCIR in Japanese-to-English direction. When the POS token was introduced, BLEU scores improved by 1.82 points on average from the baseline in Japanese-to-English translation. (ASPEC : 0.91, Tanaka : 2.73)

Furthermore, we compared proposed methods with the baseline that adopted BPE to the Japanese side only[8]. Table 3 shows the results of baseline

[6] http://www.statmt.org/moses/

[7] http://github.com/yukio326/nmt-chainer

[8] Owing to time limitations, we performed comparison with ASPEC and Tanaka corpora only, and experimented only once on each corpus.

Corpus	Baseline	Conjugation token	Conjugation feature
NTCIR	26.48%	27.43%	27.46%
ASPEC	18.56%	18.96%	18.96%
Tanaka	46.46%	53.95%	54.41%

Table 4: Vocabulary coverage.

with BPE to both English and Japanese sides. According to the results, Japanese-only BPE was inferior to the baseline without BPE.

6.2 Vocabulary coverage

The proposed method is effective in reducing the vocabulary size. The coverage of each training corpus is shown in Table 4. As for Conjugation feature, we evaluate only the number of lemmas.

It can be seen that OOV is reduced in all corpora. In particular, a significant improvement was found in the small Tanaka corpus. It can partly account for the improvement in BLEU scores in the proposed methods.

6.3 Effect of conjugation information

Experimental results showed that Conjugation token improved the BLEU score. However, Conjugation feature exhibited little or no improvement over the baselines with and without BPE. It was shown that conjugation information consists of useful features, but we should exploit the information as Conjugation token.

In the Conjugation token method, we found that the scores are influenced by the corpus size. In particular, the largest improvement was seen in a small Tanaka corpus. Conversely, Conjugation token had a small effect in a large NTCIR corpus, where both proposed methods were inferior compared to the baseline using BPE in Japanese-to-English translation. This is because the size of the corpus was sufficient to learn frequent words to produce fluent translations. Also, our method is superior to BPE in small corpus because it can compress the vocabulary without relying on frequency.

6.4 Output example

Tables 5 and 6 show the output examples in Japanese-to-English translation results.

src	彼 は 古来[10] まれ な 大 政治 家 で ある 。
ref	he is as great a statesman as ever lived .
w/o BPE	he is as great a statesman as any .
BPE	he is as great a statesman as ever lived .
C_token[9]	he is as great a statesman as ever lived .

Table 5: Output example 1.

src	これ を 下ろす[10] の てつだっ て ください 。
ref	please give me help in taking this down .
w/o BPE	please take this for me .
BPE	please take this to me .
C_token[9]	please take this down .

Table 6: Output example 2.

Table 5 depicts the handling of OOV. The baseline without BPE treated "古来" (ever lived) in this source sentence as OOV, so it could not translate the word. However, BPE and Conjugation token could translate it because it was included in each vocabulary.

Table 6 shows the handling of an unknown word. In the baseline without BPE, "下ろす" (take down) in the source sentence was represented as an unknown word token because it did not appear on the training corpus, and therefore, it failed to generate "take down" correctly. However, the conjugation token could successfully translate it because the lemma ("下ろす") which appears on the training corpus as the conditional form ("下ろせ"), continuative form ("下ろし"), and plain form ("下ろす") could be used to generate the plain form ("下ろす").

6.5 Effect of POS information

Experimental results showed that the Circumfix token (4.3) achieved the best score in Japanese-to-English translation, whereas the Conjugation token (4.1) or suffix token (4.3) was the best in English-to-Japanese translation.

We suppose that the reason for this tendency derives from the head-directionality of the target language. Because the target language in English-to-Japanese translation is Japanese, which is a head-final language, the POS token as the suffix seems to improve the translation accuracy more than the others.

However, experimental results in Japanese-to-English translation contradict this hypothesis. We assume that it is because of the right-hand head rule (Ziering and van der Plas, 2016) in English. According to this rule, basic linguistic information should be introduced before a word whereas inflection information should be placed after the word. This accounts for the different tendency in

[9] Abbreviation for Conjugation token.

[10] OOV or unknown word in the baseline.

the performance of the POS token.

7 Conclusion

In this paper, we proposed two methods using predicate conjugation information for compressing Japanese vocabulary size. The experimental results confirmed improvements in both vocabulary coverage and translation performance by using Japanese predicate conjugation information. It is important for the NMT systems to retain the grammatical property of the target language when injecting linguistic information as a special token. Moreover, it was confirmed that the proposed method is effective not only for OOV but also for unknown words.

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *Proc. of ICLR*.

Isao Goto, Ka Po Chow, Bin Lu, Eiichiro Sumita, and Benjamin K. Tsou. 2013. Overview of the patent machine translation task at the NTCIR-10 workshop. In *Proc. of NTCIR*, pages 260–286.

Thang Luong, Hieu Pham, and Christopher D. Manning. 2015a. Effective approaches to attention-based neural machine translation. In *Proc. of EMNLP*, pages 1412–1421.

Thang Luong, Ilya Sutskever, Quoc Le, Oriol Vinyals, and Wojciech Zaremba. 2015b. Addressing the rare word problem in neural machine translation. In *Proc. of ACL*, pages 11–19.

Toshiaki Nakazawa, Manabu Yaguchi, Kiyotaka Uchimoto, Masao Utiyama, Eiichiro Sumita, Sadao Kurohashi, and Hitoshi Isahara. 2016. ASPEC: Asian scientific paper excerpt corpus. In *Proc. of LREC*, pages 2204–2208.

Yasuhiro Ogawa, Mahsut Muhtar, Katsuhiko Toyama, and Yasuyoshi Inagaki. 1998. Derivational grammar approach to morphological analysis of Japanese sentences. In *Proc. of PRICAI*, pages 424–435.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a method for automatic evaluation of machine translation. In *Proc. of ACL*, pages 311–318.

Rico Sennrich and Barry Haddow. 2016. Linguistic input features improve neural machine translation. In *Proc. of WMT*, pages 83–91.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In *Proc. of ACL*, pages 1715–1725.

Patrick Ziering and Lonneke van der Plas. 2016. Towards unsupervised and language-independent compound splitting using inflectional morphological transformations. In *Proc. of NAACL*, pages 644–653.