Domain Adaptation and Attention-Based Unknown Word Replacement in Chinese-to-Japanese Neural Machine Translation

Kazuma Hashimoto, Akiko Eriguchi, and Yoshimasa Tsuruoka
The University of Tokyo, 7-3-1 Hongo, Bunkyo-ku, Tokyo, Japan
{hassy, eriguchi, tsuruoka}@logos.t.u-tokyo.ac.jp

Abstract

This paper describes our UT-KAY system that participated in the Workshop on Asian Translation 2016. Based on an Attention-based Neural Machine Translation (ANMT) model, we build our system by incorporating a domain adaptation method for multiple domains and an attention-based unknown word replacement method. In experiments, we verify that the attention-based unknown word replacement method is effective in improving translation scores in Chinese-to-Japanese machine translation. We further show results of manual analysis on the replaced unknown words.

1 Introduction

End-to-end Neural Machine Translation (NMT) with Recurrent Neural Networks (RNNs) is attracting increasing attention (Sutskever et al., 2014). By incorporating attention mechanisms (Bahdanau et al., 2015), NMT models have achieved state-of-the-art results on several translation tasks, such as English-to-German (Luong et al., 2015a) and English-to-Japanese (Eriguchi et al., 2016) tasks.

Although NMT is attractive due to its translation quality and relatively simple architecture, it is known to have some serious problems including unknown (or rare) word problems (Luong et al., 2015b). Thus, there is still room for improvement in many aspects of NMT models. In our UT-KAY system that participated in the Workshop on Asian Translation 2016 (WAT 2016) (Nakazawa et al., 2016a), we investigate the following two issues:

- adaptation with multiple domains, and
- attention-based unknown word replacement.

Our system is based on an Attention-based NMT (ANMT) model (Luong et al., 2015a). To explicitly treat translation pairs from multiple domains, our system extends a domain adaptation method for neural networks (Watanabe et al., 2016), and apply it to the baseline ANMT model. To address the unknown word problems in translated sentences, we investigate the effectiveness of replacing each unknown word according to the attention scores output by the baseline ANMT model.

In experiments, we apply our system to a Chinese-to-Japanese translation task of scientific text. Our experimental results show that the attention-based unknown word replacement method consistently improves the BLEU scores by about 1.0 for the baseline system, the domain adaptation system, and the ensemble of the two systems. Moreover, our manual analysis on the replaced unknown words indicates that the scores can be further improved if a high quality dictionary is available. While the domain adaptation method does not improve upon the baseline system in terms of the automatic evaluation metrics, the ensemble of the systems with and without the domain adaptation method boosts the BLEU score by 2.7. As a result, our UT-KAY system has been selected as one of the top three systems in the Chinese-to-Japanese task at WAT 2016.

2 Related Work

2.1 Domain Adaptation for Sentence Generation Tasks

An NMT model is usually built as a single large neural network and trained using a large parallel corpus. Such a parallel corpus is, in general, constructed by collecting sentence pairs from a variety of domains

Proceedings of the 3rd Workshop on Asian Translation,
pages 75–83, Osaka, Japan, December 11-17 2016.

(or topics), such as computer science and biomedicine. Sentences in different domains have different word distributions, and it has been shown that domain adaption is an effective way of improving image captioning models, which perform sentence generation like NMT models (Watanabe et al., 2016). Luong and Manning (2015) proposed pre-training techniques using a large general domain corpus to perform domain adaptation for NMT models. However, both of these approaches assume that there are only two domains, i.e., the source and target domains. In practice, there exist multiple topics, and thus the explicit use of information about multiple domains in the NMT models is worth investigating.

2.2 Unknown Word Replacement in NMT

Previous approaches to unknown word problems are roughly categorized into three types: character-based, subword-based, and copy-based approaches. The character-based methods aim at building word representations for unknown words by using character-level information (Luong and Manning, 2016). The character-based methods can handle any words and has achieved better results than word-based methods. However, the computational cost grows rapidly.

Recently, Sennrich et al. (2016) have shown that the use of subword units in NMT models is effective. The subword units can treat multiple levels of granularity existing in words and reduce the size of the vocabulary compared with the standard word-based models. However, the rules to use the subword units are built based on the training data, and thus there still remains the problem of treating an infinite number of the unknown words.

The copy-based methods aim at copying relevant source words to replace unknown words. Some use existing alignment tools (Luong et al., 2015b), and others suggest that using attention scores in the ANMT models can be an alternative to using alignment tools (Jean et al., 2015). The copy-based method should be effective in translation tasks where characters and words are shared across different languages. However, to the best of our knowledge, there is no previous work which inspects the replacement results based on the attention mechanism to investigate the relevance of the replacement method in ANMT models.

3 The UT-KAY System at WAT 2016

In this section, we describe our UT-KAY system at WAT 2016. We first describe the baseline method for our system in Section 3.1 and then explain how our system works in Section 3.2, 3.3, and 3.4.

3.1 Baseline Methods

3.1.1 Attention-Based Sequential NMT

We employ an ANMT model presented in Luong et al. (2015a) as our baseline and follow the single-layer setting as in Eriguchi et al. (2016). Let us represent the source sentence of length M by word sequence $\boldsymbol{x} = (x_1, x_2, \dots, x_M)$ and its corresponding target sentence of length N by word sequence $\boldsymbol{y} = (y_1, y_2, \dots, y_N)$. For embedding the source word sequence $\boldsymbol{x}$ into a d_h-dimensional vector space, Long Short-Term Memory (LSTM) units are used as follows:

$$\boldsymbol{s}_i = \mathrm{LSTM}(\boldsymbol{s}_{i-1}, \boldsymbol{v}(x_i)), \tag{1}$$

where $\boldsymbol{s}_i$ and $\boldsymbol{s}_{i-1} \in \mathbb{R}^{d_h \times 1}$ are the i-th and $(i-1)$-th hidden states, $\boldsymbol{s}_0$ is filled with zeros, and $\boldsymbol{v}(x_i) \in \mathbb{R}^{d_e \times 1}$ is the d_e-dimensional word embedding of the i-th source word x_i. The LSTM function is formulated with internal states, called *memory cells*, as follows:

$$\begin{aligned}
\boldsymbol{i}_i &= \sigma(\boldsymbol{U}_s^{\boldsymbol{i}} \boldsymbol{s}_{i-1} + \boldsymbol{V}_s^{\boldsymbol{i}} \boldsymbol{v}(x_i) + \boldsymbol{b}_s^{\boldsymbol{i}}), & \boldsymbol{f}_i &= \sigma(\boldsymbol{U}_s^{\boldsymbol{f}} \boldsymbol{s}_{i-1} + \boldsymbol{V}_s^{\boldsymbol{f}} \boldsymbol{v}(x_i) + \boldsymbol{b}_s^{\boldsymbol{f}}), \\
\boldsymbol{o}_i &= \sigma(\boldsymbol{U}_s^{\boldsymbol{o}} \boldsymbol{s}_{i-1} + \boldsymbol{V}_s^{\boldsymbol{o}} \boldsymbol{v}(x_i) + \boldsymbol{b}_s^{\boldsymbol{o}}), & \boldsymbol{u}_i &= \tanh(\boldsymbol{U}_s^{\boldsymbol{u}} \boldsymbol{s}_{i-1} + \boldsymbol{V}_s^{\boldsymbol{u}} \boldsymbol{v}(x_i) + \boldsymbol{b}_s^{\boldsymbol{u}}), \\
\boldsymbol{c}_i &= \boldsymbol{i}_t \odot \boldsymbol{u}_i + \boldsymbol{f}_i \odot \boldsymbol{c}_{i-1}, & \boldsymbol{s}_i &= \boldsymbol{o}_i \odot \tanh(\boldsymbol{c}_i),
\end{aligned} \tag{2}$$

where $\boldsymbol{U}_s \in \mathbb{R}^{d_h \times d_h}$, $\boldsymbol{V}_s \in \mathbb{R}^{d_h \times d_e}$, $\boldsymbol{b}_s \in \mathbb{R}^{d_h \times 1}$ are the LSTM's weight matrices and bias vectors, and $\boldsymbol{c}_i \in \mathbb{R}^{d_h \times 1}$ is the memory cell. The operator $\odot$ denotes element-wise multiplication and $\sigma(\cdot)$ is the logistic sigmoid function.

Once $\boldsymbol{s}_M$, which represents the entire source sentence $\boldsymbol{x}$, is obtained for representing the source sentence $\boldsymbol{x}$, the ANMT model estimates the conditional probability that the j-th target word y_j is generated given the target word sequence $(y_1, y_2, \dots, y_{j-1})$ and the source sentence $\boldsymbol{x}$:

$$p(y_j|y_1, y_2, \dots, y_{j-1}, \boldsymbol{x}) = \mathrm{softmax}(\boldsymbol{W}_p \tilde{\boldsymbol{t}}_j + \boldsymbol{b}_p), \tag{3}$$

where $\boldsymbol{W}_p \in \mathbb{R}^{|\mathbb{V}_t| \times d_h}$ and $\boldsymbol{b}_p \in \mathbb{R}^{|\mathbb{V}_t| \times 1}$ are an weight matrix and a bias vector, $\mathbb{V}_t$ is the target word vocabulary, and $\tilde{\boldsymbol{t}}_j \in \mathbb{R}^{d_h \times 1}$ is the hidden state for generating the j-th target word. In general, the target word vocabulary $\mathbb{V}_t$ is constructed by a pre-defined number of the most frequent words in the training data, and the other words are mapped to a special token *UNK* to indicate that they are unknown words. $\tilde{\boldsymbol{t}}_j$ is conditioned by the j-the hidden state $\boldsymbol{t}_j \in \mathbb{R}^{d_h \times 1}$ of another LSTM RNN and an attention vector $\boldsymbol{a}_j \in \mathbb{R}^{d_h \times 1}$ as follows:

$$\boldsymbol{t}_j = \mathrm{LSTM}(\boldsymbol{t}_{j-1}, [\boldsymbol{v}(y_{j-1}); \tilde{\boldsymbol{t}}_{j-1}]), \tag{4}$$

$$\boldsymbol{a}_j = \sum_{i=1}^{M} \alpha_{(j,i)} \boldsymbol{s}_i, \tag{5}$$

$$\tilde{\boldsymbol{t}}_j = \tanh(\boldsymbol{W}_t \boldsymbol{t}_j + \boldsymbol{W}_a \boldsymbol{a}_j + \boldsymbol{b}_{\tilde{t}}), \tag{6}$$

where $[\boldsymbol{v}(y_{j-1}); \tilde{\boldsymbol{t}}_{j-1}] \in \mathbb{R}^{(d_e+d_h) \times 1}$ is the concatenation of $\boldsymbol{v}(y_{j-1})$ and $\tilde{\boldsymbol{t}}_{j-1}$, and $\boldsymbol{W}_t \in \mathbb{R}^{d_h \times d_h}$, $\boldsymbol{W}_a \in \mathbb{R}^{d_h \times d_h}$, and $\boldsymbol{b}_{\tilde{t}} \in \mathbb{R}^{d_h \times 1}$ are weight matrices and a bias vector. To use the information about the source sentence, $\boldsymbol{t}_1$ is set equal to $\boldsymbol{s}_M$. The attention score $\alpha_{(j,i)}$ is used to estimate how important the i-th source-side hidden state $\boldsymbol{s}_i$ is, for predicting the j-the target word:

$$\alpha_{(j,i)} = \frac{\exp(\boldsymbol{t}_j \cdot \boldsymbol{s}_i)}{\sum_{k=1}^{M} \exp(\boldsymbol{t}_j \cdot \boldsymbol{s}_k)}, \tag{7}$$

where $\boldsymbol{t}_j \cdot \boldsymbol{s}_k$ is the dot-product used to measure the relatedness between the two vectors.

All of the model parameters in the ANMT model are optimized by minimizing the following objective function:

$$J(\boldsymbol{\theta}) = -\frac{1}{|\mathcal{T}|} \sum_{(\boldsymbol{x},\boldsymbol{y}) \in \mathcal{T}} \sum_{j=1}^{N} \log p(y_j|y_1, y_2, \dots, y_{j-1}, \boldsymbol{x}), \tag{8}$$

where $\boldsymbol{\theta}$ denotes the set of the model parameters, and $\mathcal{T}$ is the set of source and target sentence pairs to train the model.

3.1.2 Domain Adaptation for Neural Networks by Feature Augmentation

To learn translation pairs from multiple domains, we employ a domain adaptation method which has proven to be effective in neural image captioning models (Watanabe et al., 2016). It should be noted that neural image captioning models can be formulated in a similar manner to NMT models. That is, if we use images as source information instead of the source sentences, the task is then regarded as an image captioning task. We use the corresponding equations in Section 3.1.1 to describe the domain adaptation method by assuming that $\boldsymbol{x}$ is a representation of an image.

The method assumes that we have data from two domains: $\mathcal{D}_1$ and $\mathcal{D}_2$. The softmax parameters $(\boldsymbol{W}_p, \boldsymbol{b}_p)$ in Equation (3) are separately assigned to each of the two domains, and the parameters $(\boldsymbol{W}_p^{\mathcal{D}_1}, \boldsymbol{b}_p^{\mathcal{D}_1})$ and $(\boldsymbol{W}_p^{\mathcal{D}_2}, \boldsymbol{b}_p^{\mathcal{D}_2})$ are decomposed into two parts as follows:

$$\boldsymbol{W}_p^{\mathcal{D}_1} = \boldsymbol{W}_p^{\mathcal{G}} + \overline{\boldsymbol{W}}_p^{\mathcal{D}_1}, \quad \boldsymbol{b}_p^{\mathcal{D}_1} = \boldsymbol{b}_p^{\mathcal{G}} + \overline{\boldsymbol{b}}_p^{\mathcal{D}_1}, \quad \boldsymbol{W}_p^{\mathcal{D}_2} = \boldsymbol{W}_p^{\mathcal{G}} + \overline{\boldsymbol{W}}_p^{\mathcal{D}_2}, \quad \boldsymbol{b}_p^{\mathcal{D}_2} = \boldsymbol{b}_p^{\mathcal{G}} + \overline{\boldsymbol{b}}_p^{\mathcal{D}_2}, \tag{9}$$

where $(\boldsymbol{W}_p^{\mathcal{G}}, \boldsymbol{b}_p^{\mathcal{G}})$ is the shared component in $(\boldsymbol{W}_p^{\mathcal{D}_1}, \boldsymbol{b}_p^{\mathcal{D}_1})$ and $(\boldsymbol{W}_p^{\mathcal{D}_2}, \boldsymbol{b}_p^{\mathcal{D}_2})$, and $(\overline{\boldsymbol{W}}_p^{\mathcal{D}_1}, \overline{\boldsymbol{b}}_p^{\mathcal{D}_1})$ and $(\overline{\boldsymbol{W}}_p^{\mathcal{D}_2}, \overline{\boldsymbol{b}}_p^{\mathcal{D}_2})$ are the domain-specific components. Intuitively, the shared component learns general information across multiple domains, and the domain-specific components learn domain-specific information.

The model parameters are optimized by replacing the negative log-likelihood in Equation (8) as follows:

$$-\frac{1}{2}\log p^{\mathcal{G}}(y_j|y_1, y_2, \ldots, y_{j-1}, \boldsymbol{x}) - \frac{1}{2}\log p^{\mathcal{D}_1}(y_j|y_1, y_2, \ldots, y_{j-1}, \boldsymbol{x}), \tag{10}$$

$$-\frac{1}{2}\log p^{\mathcal{G}}(y_j|y_1, y_2, \ldots, y_{j-1}, \boldsymbol{x}) - \frac{1}{2}\log p^{\mathcal{D}_2}(y_j|y_1, y_2, \ldots, y_{j-1}, \boldsymbol{x}), \tag{11}$$

where the first one is used for data from $\mathcal{D}_1$, and the second one is used for data from $\mathcal{D}_2$. The probabilities $p^{\mathcal{G}}, p^{\mathcal{D}_1}, p^{\mathcal{D}_2}$ are computed by replacing $(\boldsymbol{W}_p, \boldsymbol{b}_p)$ in Equation (3) with $(2\boldsymbol{W}_p^{\mathcal{G}}, 2\boldsymbol{b}_p^{\mathcal{G}})$, $(2\overline{\boldsymbol{W}}_p^{\mathcal{D}_1}, 2\overline{\boldsymbol{b}}_p^{\mathcal{D}_1})$, $(2\overline{\boldsymbol{W}}_p^{\mathcal{D}_2}, 2\overline{\boldsymbol{b}}_p^{\mathcal{D}_2})$, respectively. At test time, we only use $(\boldsymbol{W}_p^{\mathcal{D}_1}, \boldsymbol{b}_p^{\mathcal{D}_1})$ and $(\boldsymbol{W}_p^{\mathcal{D}_2}, \boldsymbol{b}_p^{\mathcal{D}_2})$ to compute the output probabilities.

3.2 Adaptation with Multiple Domains

The domain adaptation method described in Section 3.1.2 assumes that we have only two domains, namely, source and target domains, but in practice we can have many domains. In this paper, we extend the domain adaptation method in order to treat data from multiple domains. Assuming that we have data from K domains $\mathcal{D}_1, \mathcal{D}_2, \ldots, \mathcal{D}_K$, we decompose the softmax parameters for the K domains in exactly the same way as Equation (9). That is, we have $K + 1$ softmax parameters: $(\boldsymbol{W}_p^{\mathcal{G}}, \boldsymbol{b}_p^{\mathcal{G}}), (\overline{\boldsymbol{W}}_p^{\mathcal{D}_1}, \overline{\boldsymbol{b}}_p^{\mathcal{D}_1}), \ldots, (\overline{\boldsymbol{W}}_p^{\mathcal{D}_K}, \overline{\boldsymbol{b}}_p^{\mathcal{D}_K})$.

The reformulated objective function in the domain adaptation method can be optimized in the same way as the ANMT model, and to speed up the training, we apply *BlackOut* (Ji et al., 2016), a sampling-based approximation method for large vocabulary language modeling, as in Eriguchi et al. (2016). More specifically, we independently approximate the output probabilities $p^{\mathcal{G}}, p^{\mathcal{D}_1}, \ldots, p^{\mathcal{D}_K}$ using BlackOut sampling. To sample negative examples, we use the global unigram distribution computed by using all data from all the K domains.

3.3 Attention-Based Unknown Word Replacement

As described in Section 3.1.1, the ANMT model computes an attention score $\alpha_{(j,i)}$ to estimate how important the i-th hidden state in the source-side LSTM is for generating the j-th target word. Although the attention mechanism is not designed as word alignment in traditional statistical machine translation, it is observed that high attention scores are often assigned to word-level translation pairs (Bahdanau et al., 2015).

In this paper, we investigate the effectiveness of using the attention scores to replace unknown words in translated sentences with words in source sentences. For each unknown word *UNK* in a translated sentence, we replace it with the source word having the highest attention score. That is, we replace the j-th target word (*UNK*) with the k-th source word when $\alpha_{(j,k)}$ is the highest among $\alpha_{(j,i)}$ $(1 \leq i \leq M)$.

3.4 Ensemble

At test time, we employ two separate ensemble techniques.

3.4.1 Ensemble of Output Probabilities

The first one is a widely-used technique which takes the average of output probabilities for generating target words. Here we can treat each NMT model as a black box to output a word probability distribution and take the average of the probabilities from all of the models:

$$\frac{1}{L}\sum_{k=1}^{L} p_k(y_j|y_1, y_2, \ldots, y_{j-1}, \boldsymbol{x}), \tag{12}$$

where p_k is the probability from the k-th NMT model, and L is the number of NMT models we have.

3.4.2 Ensemble of Attention Scores

When generating target sentences using the ensemble technique described in Section 3.4.1, we have L set of attention scores, assuming that we use the ANMT models. We use the averaged attention scores over the L set of attention scores each time we replace an unknown word by the method described in Section 3.3. It should be noted that the two ensemble techniques in this section are separately used and thus the attention scores are *not* averaged during the sentence generation step.

4 Experimental Settings

4.1 Data

At WAT 2016, there are several tasks for several language pairs. In this work, we choose the ASPEC Chinese-to-Japanese (ASPEC-CJ) translation task[1] (Nakazawa et al., 2016b) as our first step to investigate the effectiveness of our system. The ASPEC-CJ dataset includes sentences from multiple domains with annotations, and the language pair shares the same Chinese characters. There are 10 predefined domain tags: *Abs, AGS, BIO, CHEM, ENE, ENVI, INFO, MATE, MED, SP*. We treated Abs, BIO, and MED as a single domain, and also INFO and SP.[2] Consequently, the number of the domains was 7 in our experiments.

The training data includes 672,315 sentence pairs, the development data includes 2,090 sentence pairs, and the test data includes 2,107 sentence pairs. We used all of the training data to train the ANMT models, and built the source and target vocabularies using the words which appear in the training data more than twice. Consequently, the source vocabulary has 94,961 words and the target vocabulary has 69,580 words, including the UNK token and a special token for representing the end of sentences *EOS* for each vocabulary. The source and target words were obtained using the Kytea tool and the Stanford Core NLP tool, respectively.[3]

4.2 Parameter Optimization and Translation

We set the dimensionality of word embeddings and hidden states to 512 (i.e., $d_e = d_h = 512$), and we used single-layer LSTMs. All of the model parameters, except for the bias vectors and weight matrices of the softmax layer, were initialized with uniform random values from $[-1.0, 1.0]$. The bias vectors and the softmax weight were initialized with zeros. For the BlackOut approximation method, we set the number of negative samples to 2,000, and the objective function was optimized via mini-batch stochastic gradient descent. The mini-batch size was 128 without any constraints, such as lengths of the sentences in each mini-batch. All of the mini-batches were constructed randomly at the beginning of each epoch. For each mini-batch, the gradients were divided by the mini-batch size, and then the L2 norm of the gradients was clipped to 3. The initial learning rate was 1.0 and it was multiplied by 0.75 to decrease the learning rate when the perplexity for the development data did not improve. We checked the perplexity for the development data after each epoch.[4]

To generate the target sentences, we used the beam search strategy based on the statistics of the sentence lengths as in Eriguchi et al. (2016). We set the beam size to 20, and after generating the sentences using the beam search, we applied the attention-based unknown word replacement method to the unknown words output by the system.

5 Results and Discussion

5.1 Main Results

Table 1 shows our experimental results in terms of BLEU and RIBES scores for the development and test data. In the table, the results of the best systems at WAT 2015 and WAT 2016, Neubig et al. (2015) and Kyoto-U, are also shown. These results are the Kytea-based evaluation results. First, we can see

[1] `http://lotus.kuee.kyoto-u.ac.jp/ASPEC/`

[2] The categorization was based on personal communication with the organizer of WAT 2016.

[3] `http://www.phontron.com/kytea/` and `http://stanfordnlp.github.io/CoreNLP/`.

[4] Our system was implemented using our CPU-based neural network library: `https://github.com/hassyGo/N3LP`.

	Dev. data		Test data	
Method	BLEU	RIBES	BLEU	RIBES
(1) ANMT	38.09	83.67	-	-
(2) ANMT w/ UNK replacement	39.05	83.98	39.06	84.23
(3) ANMT w/ domain adaptation	38.28	83.83	-	-
(4) ANMT w/ domain adaptation and UNK replacement	39.24	84.20	39.07	84.21
(5) Ensemble of (1) and (3)	40.66	84.91	-	-
(6) Ensemble of (1) and (3) w/ UNK replacement	41.72	85.25	41.81	85.47
The best system at WAT 2015 (Neubig et al., 2015)	-	-	42.95	84.77
The best system at WAT 2016 (Kyoto-U, NMT)	-	-	46.70	87.29

Table 1: Kytea-based BLEU and RIBES scores for the development and test data on the ASPEC-CJ task. The results (4) and (6) were submitted to the official evaluation system.

that the attention-based unknown word replacement method (*UNK replacement* in the table) consistently improves the BLEU scores by about 1.0, and the RIBES scores by about 0.3 for the development data. Next, currently we have not observed significant improvement by using the domain adaptation method, in terms of the BLEU and RIBES scores. Finally, the ensemble of the two ANMT models with and without domain adaptation consistently improves the translation scores, and in particular, the BLEU score improves by about 2.7, and the RIBES score improves by about 1.2. In most of previous work on NMT models, the ensemble is performed by using exactly the same models with different parameter initialization settings. By contrast, we performed the ensemble using two different models with different objective functions, and observed large gains for the BLEU scores.

The Kyoto-U system achieved the best results at WAT 2016, and it is also based on NMT. The system's scores are much better than ours. As shown in the system description on the official website, the system seems to be based on sophisticated methods, such as "reverse scoring", which should be helpful in improving our system. In general, results of NMT models highly depend not only on such techniques, but also on their model settings, such as the number of RNN layers and dimensionality of embeddings and hidden states. Thus, it is not surprising that the two NMT-based systems produce such different scores.

5.2 Analysis on Attention-Based Unknown Word Replacement

To inspect the results of the attention-based unknown word replacement method, we manually checked the translated sentences of the development data. In the translated sentences in our best result by the method (6), we found 690 sentences which include unknown words. Among them, we sampled 132 sentences including 250 unknown words. Then we categorized all the cases into five types as follows:

(A) Correct A replacement case is categorized as (A) if the replaced word is picked up from its relevant position in its source sentence and exactly the same as the corresponding word in its reference translation. Thus, type (A) contributes to improving BLEU scores.

(B) Acceptable A replacement case is categorized as (B) if the replaced word is picked up from its relevant position, but it is not the same as the reference word while it fits the translated Japanese sentence. That is, type (B) is semantically acceptable, but it does not contribute to improving BLEU scores.

(C) Correct with word translation A replacement case is categorized as (C) if the replaced word is picked up from its relevant position, but it is a Chinese word which should be translated into its corresponding Japanese words.

(D) Partially correct A replacement case is categorized as (D) if the replaced word is picked up from its relevant position but some words are missing. Thus, it cannot be regarded as a sufficient translation.

Type	Count	Ratio
(A) Correct	76	30.4%
(B) Acceptable	5	2.0%
(C) Correct with word translation	104	41.6%
(D) Partially correct	50	20.0%
(E) Incorrect	15	6.0%
Total	250	100.0%

Table 2: Analysis on the attention-based unknown word replacement method for 250 replacements in 132 translated sentences of the development data.

R. 1	Yukon や北西領域，Hudson や James 湾，北部ケベック，ラブラドール， グリーンランドの汚染物質に関する情報を，文献，組織，研究者から広範囲に収集した。
T. 1	Yukon$_{\text{(A)}}$ と北西分野，Hudson$_{\text{(A)}}$ と James$_{\text{(A)}}$ 湾，北部の 魁北克$_{\text{(C)}}$，拉布拉多$_{\text{(C)}}$， Greenland$_{\text{(B)}}$ の汚染物質の情報について文献，組織，研究者から広範囲の収集を行った。
R. 2	高尾山の環境保全と京王の社会貢献
T. 2	高 尾山$_{\text{(A)}}$ の環境保全と 京$_{\text{(D)}}$ の社会貢献

Table 3: Examples of the unknown word replacements.

(E) Incorrect A replacement case is categorized as (E) if the replaced word is picked up from an irrelevant position and it does not make sense in its translated sentence.

Table 2 shows the results of the manual analysis on the 250 cases. We can see that about 30% of the unknown word replacements are categorized as (A), which leads to the improvement of the BLEU score by 1.06 (40.66→41.72) in Table 1. The majority is type (C), and thus it is still room for improvement in the results by combining external resources like word-level dictionaries.[5] These results suggest that the attention-based unknown word replacement method can be a simple way for improving translation results in Chinese-to-Japanese translation, and the method can be used in any attention-based NMT models.

Table 3 shows two examples of the translated sentences which include unknown words, and for each example, its reference translation (**R.**) and its translation result (**T.**) are shown. The replaced unknown words are underlined with their corresponding replacement types. In the first example, there are six unknown words, and all of them are categorized as (A), (B), or (C), which means that the ANMT model can distinguish between different unknown words even though all of them are represented with the special token *UNK*. The replaced Chinese word "魁北克" means "Quebec" and "ケベック" in English and Japanese, respectively, and "拉布拉多" means "Labrador" and "ラブラドール" in English and Japanese, respectively. The two replacements are categorized as (C) because they need to be translated into their corresponding Japanese words. The word "Greenland" means "グリーンランド" in Japanese, and it seems that some English words are also used in the reference sentences. Thus we categorized this case as (B).

In the second example, there are two unknown words, and both of them are related to named entities; "高尾山" is a Japanese mountain and "京王" is a Japanese company. However, as opposed to our expectation, "高尾山" is split into "高" and "尾山", and "京王" is split into "京" and "王". As a result, the unknown word replacement method picks up only a part of the word "京王", which leads to an insufficient translation (categorized as (D)). These results suggest that improving the accuracy of the word segmentation will lead to better translation results by the ANMT models.

5.3 Analysis on Domain Adaptation

We inspected the BLEU scores for development data of each domain. Tabe 4 shows the results of the methods (2), (4), and (6) presented in Table 1 and the number of sentence pairs for each domain. From

[5] We tried to automatically build a dictionary using a word alignment tool, but the word segmentation results were so noisy that we could not obtain informative dictionary.

		BIO	CHEM	ENE	ENVI	INFO	MATE
BLEU	Method (2)	35.27	37.24	39.74	36.21	41.91	34.92
	Method (4)	34.86	33.96	40.37	37.16	41.58	37.80
	Method (6)	37.84	42.77	43.64	39.29	44.17	38.65
# of samples in the development data		216	19	37	804	982	32

Table 4: BLEU scores for the development data of each domain.

these results we can see that the domain adaptation method (Method (4)) performs better than the baseline method (Method (2)) in some domains, but not in others. The ensemble result (Method (6)) consistently improves the results for all of the domains.

We expect the domain adaptation method to disambiguate the meaning of a word according to its context. For example in Table 3, both of the Japanese words "領域" and "分野" mean "field" and "area" but their meanings depend on their context. In such a case, the domain or context information should be helpful in disambiguating the meanings. However, none of our methods could successfully output the appropriate word "領域". To investigate the result, we inspected the usage of the Japanese word " 領域" in the training data, and found that similar usages to the above example were rare. Therefore, this *rare usage problem* would be addressed, not by the domain adaptation method, but by adding large monolingual data to make the language modeling more accurate.

6 Conclusion

This system description paper presented our UT-KAY system based on an attention-based neural machine translation model. We investigated the effectiveness of a domain adaptation method and an attention-based unknown word replacement method. The domain adaptation method does not currently lead to better results than our baseline model. By contrast, we have found that the attention-based unknown word replacement has potential benefits in Chinese-to-Japanese NMT models, which can be applied to any attention-based models.

Acknowledgments

This work was supported by CREST, JST.

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural Machine Translation by Jointly Learning to Align and Translate. In *Proceedings of the 3rd International Conference on Learning Representations*.

Akiko Eriguchi, Kazuma Hashimoto, and Yoshimasa Tsuruoka. 2016. Tree-to-Sequence Attentional Neural Machine Translation. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 823–833.

Sébastien Jean, Orhan Firat, Kyunghyun Cho, Roland Memisevic, and Yoshua Bengio. 2015. Montreal Neural Machine Translation Systems for WMT ' 15. In *Proceedings of the Tenth Workshop on Statistical Machine Translation*, pages 134–140.

Shihao Ji, S. V. N. Vishwanathan, Nadathur Satish, Michael J. Anderson, and Pradeep Dubey. 2016. BlackOut: Speeding up Recurrent Neural Network Language Models With Very Large Vocabularies. In *Proceedings of the 4th International Conference on Learning Representations*.

Minh-Thang Luong and Christopher D. Manning. 2015. Stanford Neural Machine Translation Systems for Spoken Language Domains. In *Proceedings of the 12th International Workshop on Spoken Language Translation*, pages 76–79.

Minh-Thang Luong and Christopher D. Manning. 2016. Achieving Open Vocabulary Neural Machine Translation with Hybrid Word-Character Models. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1054–1063.

Minh-Thang Luong, Hieu Pham, and Christopher D. Manning. 2015a. Effective Approaches to Attention-based Neural Machine Translation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1412–1421.

Minh-Thang Luong, Ilya Sutskever, Quoc Le, Oriol Vinyals, and Wojciech Zaremba. 2015b. Addressing the Rare Word Problem in Neural Machine Translation. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 11–19.

Toshiaki Nakazawa, Hideya Mino, Chenchen Ding, Isao Goto, Graham Neubig, Sadao Kurohashi, and Eiichiro Sumita. 2016a. Overview of the 3rd Workshop on Asian Translation. In *Proceedings of the 3rd Workshop on Asian Translation (WAT2016)*.

Toshiaki Nakazawa, Manabu Yaguchi, Kiyotaka Uchimoto, Masao Utiyama, Eiichiro Sumita, Sadao Kurohashi, and Hitoshi Isahara. 2016b. Aspec: Asian scientific paper excerpt corpus. In *Proceedings of the 10th Conference on International Language Resources and Evaluation (LREC2016)*.

Graham Neubig, Makoto Morishita, and Satoshi Nakamura. 2015. Neural Reranking Improves Subjective Quality of Machine Translation: NAIST at WAT2015. In *Proceedings of the 2nd Workshop on Asian Translation (WAT2015)*, pages 35–41.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural Machine Translation of Rare Words with Subword Units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to Sequence Learning with Neural Networks. In *Advances in Neural Information Processing Systems 27*, pages 3104–3112.

Yusuke Watanabe, Kazuma Hashimoto, and Yoshimasa Tsuruoka. 2016. Domain Adaptation for Neural Networks by Parameter Augmentation. In *Proceedings of the 1st Workshop on Representation Learning for NLP*, pages 249–257.