LIT Team's System Description for Japanese-Chinese Machine Translation Task in IWSLT 2020

Yimeng Zhuang, Yuan Zhang, Lijie Wang
Samsung Research China - Beijing (SRC-B)
{ym.zhuang, yuan.zhang, lijie.wang}@samsung.com

Abstract

This paper describes the LIT Team's submission to the IWSLT2020 open domain translation task, focusing primarily on Japanese-to-Chinese translation direction. Our system is based on the organizers' baseline system, but we do more works on improving the Transformer baseline system by elaborate data preprocessing. We manage to obtain significant improvements, and this paper aims to share some data processing experiences in this translation task. Large-scale back-translation on monolingual corpus is also investigated. In addition, we also try shared and exclusive word embeddings, compare different granularity of tokens like sub-word level. Our Japanese-to-Chinese translation system achieves a performance of BLEU=34.0 and ranks 2nd among all participating systems.

1 Introduction

In recent years, the neural machine translation (NMT) (Sun et al., 2019; Wu et al., 2016; Sennrich et al., 2015) has made great progress based on encoder-decoder architecture. We participate in the IWSLT 2020 open domain translation task: Japanese-to-Chinese. This paper describes the NMT systems for the IWSLT 2020 Japanese-to-Chinese machine translation task (Ansari et al., 2020).

Our main efforts are data pre-processing, specifically parallel data filter and sentence alignment. By elaborate data processing, we successfully improve the quality of the training set and thus boost the performance of our translation system. The back-translation mechanism (Edunov et al., 2018) is also investigated to extend the training corpus, we translate Chinese to Japanese to get the Japanese-to-Chinese training corpus, it is an effective approach to exploit the corresponding monolingual data sets.

The transformer model (Vaswani et al., 2017) based on multi-head attention has achieved excellent performance in a variety of neural machine translation tasks in the last three years. This kind of NMT model surpasses the performance of the traditional statistical machine translation and the NMT performs particularly well especially with rich resource corpus. In our system, we adopt bigger transformer architecture, since the performance of the Transformer relies on model capacity, ex. the number of dimensions of the feed-forward network. To improve performance, we adopted the Relative Position Attention (Shaw et al., 2018). Also, we conduct experiments to compare the shared source and target word embeddings and exclusive word embeddings, and whether to adopt the shared embeddings relates to the translation direction. The Chinese-to-Japanese direction achieves higher scores when adopting shared word embeddings, however, the Japanese-to-Chinese direction produces opposite results.

The paper is structured as follows: Section 2 will present a detailed description of our data preprocessing, back-translation is introduced in Section 3, the model of our system will be introduced in Section 4, the main results of our experiment will be shown in Section 5. Section 6 will draw a brief conclusion of our work for the IWSLT 2020 open domain translation task.

2 Data and Pre-processing

On the whole, our system follows the standard Transformer-based translation pipeline, and our system implementation is based on the official baseline [1]. Most of our efforts in this competition are focused on data pre-processing and back-translation. We adopt the same strategies as the official baseline if we don't point out explicitly.

[1] https://github.com/didi/iwslt2020_open_domain_translation

Proceedings of the 17th International Conference on Spoken Language Translation (IWSLT), pages 109–113
July 9-10, 2020.
https://doi.org/10.18653/v1/P17

Corpus	Original	Re-Filtered
Part A	2.0M	0.8M
Part B	19.0M	8.8M
Part C	161.5M	10.0M
Part D	-	2.7M
Monolingual	-	200M

Table 1: Statistics of the parallel sentence pairs used for training models in this paper. The monolingual data is used for back-translation.

2.1 Datasets

In Japanese-Chinese bidirectional machine translation competition(Ansari et al., 2020), the organizers provide a large, noisy set of Japanese-Chinese segment pairs built from web data. There are four parts:

Part A A small but relatively clean Japanese-Chinese parallel corpus, which is obtained from curating existing Japanese-Chinese parallel datasets.

Part B A pre-filtered dataset of the sentences that the organizers obtained from crawling the Web, aligning, and filtering.

Part C An unfiltered parallel web crawled corpus which is much noisier than previous datasets.

Part D A huge file of the unaligned scraped web pages with the document boundaries.

The following subsections detail how we handle these four parts of web data respectively. Besides, we conduct data augmentation by back-translation using extra monolingual data, which will be discussed in Section 3. Table 1 shows the statistics of the training data.

2.2 Parallel Data Filter

Although the organizers have filtered parts of the data, there are still many mismatched sentence pairs, i.e. the target sentence is not the corresponding translation of the source sentence. For the three aligned datasets, we re-filter them by the following rules.

- Remove empty or duplicated sentences.
- Remove sentence pairs when the source sentence and the target sentence are same.
- Convert all Chinese characters into simplified Chinese.
- Remove sentence pairs when there is no common Chinese character (Chu et al., 2014) between source sentence and target sentence.
- Remove sentences in which the number of non-English and non-punctuation characters is less than half of the length of the whole sentence.
- The maximum length ratio of sentence pairs is 1.8.
- Japanese (Chinese) sentence should be recognized as Japanese (Chinese) by fasttext's language identification model (Joulin et al., 2016b,a).

After that, we have a pre-processed bilingual training data consisting of 22.3 million parallel sentences. Note that we adjust filter rules many times, and finally adopt the above relatively strict rules, resulting in the training data is reduced significantly. Besides, the `fast_align` [2] toolkit is popular for computing the alignment score for parallel sentences, but the models trained on the training data filtered by `fast_align` become worse. We suspect that its scores may be highly related to the length of the sentences, this results in qualified long sentences are discarded. So we didn't use `fast_align` in this paper.

2.3 Web Crawled Sentence Alignment

The organizers provide us a huge corpus of more than 15 million unaligned bilingual document pairs. To extract the parallel sentences, we consider each sentence of a document as an element and adopt the longest common sub-sequence algorithm to find Ja-Zh sentence pairs with the highest character F1 similarity. Algorithm 1 shows the alignment process, in which we define the alignment score $score(C_i, J_j)$ between two sentences by the F1 value of their character overlap.

Unfortunately, this part of data is highly duplicated. After performing Algorithm 1 and filter algorithm mentioned in Section 2.2, we successfully obtained about 50 million parallel sentences pairs. But only 2.7 million sentence pairs are remained after deduplicating. Something is better than nothing, we still add the 2.7 million data into our training set.

3 Back-translation

In recent works, the back-translation mechanism (Edunov et al., 2018) has been proved as an effective method to improve machine translation systems by utilizing large-scale monolingual corpus.

[2] https://github.com/clab/fast_align

Algorithm 1 Align bilingual sentences from two documents.

Require: Chinese sentences $C_1, C_2, \cdots, C_N$; Japanese sentences $J_1, J_2, \cdots, J_M$;
Ensure: Aligned sentence pairs set A

Initialize all auxiliary variables s to zero;
for $i = 1 \rightarrow N$ **do**
  for $j = 1 \rightarrow M$ **do**
    if $s_{i-1,j} \geq s_{i,j}$ **then**
      $s_{i,j} \leftarrow s_{i-1,j}, trace(i,j) \leftarrow 0$
    end if
    if $s_{i,j-1} \geq s_{i,j}$ **then**
      $s_{i,j} \leftarrow s_{i,j-1}, trace(i,j) \leftarrow 1$
    end if
    if $s_{i-1,j-1} \geq s_{i,j}$ **then**
      $s_{i,j} \leftarrow s_{i-1,j-1}, trace(i,j) \leftarrow 2$
    end if
    if $s_{i-1,j-1} + score(C_i, J_j) > s_{i,j}$ **then**
      $s_{i,j} \leftarrow s_{i-1,j-1} + score(C_i, J_j)$
      $trace(i,j) \leftarrow 3$
    end if
  end for
end for
$i \leftarrow N, j \leftarrow M$
while $i > 0$ and $j > 0$ **do**
  if $trace(i,j) = 0$ **then**
    $i \leftarrow i - 1$
  else if $trace(i,j) = 1$ **then**
    $j \leftarrow j - 1$
  else if $trace(i,j) = 2$ **then**
    $i \leftarrow i - 1, j \leftarrow j - 1$
  else if $trace(i,j) = 3$ **then**
    add sentence pair (C_i, J_j) to set A
    $i \leftarrow i - 1, j \leftarrow j - 1$
  end if
end while

In this paper, we follow the successful experiences in Edunov et al. (2018) to further extend our training data. Chinese monolingual data is extracted from the unaligned scraped web pages (Part D), and we select 200 million sentences to reduce training time.

3.1 Chinese-to-Japanese Translation

In order to generate a synthetic bilingual corpus, we trained a Chinese-to-Japanese transformer on the filtered parallel data mentioned in Section 2.2. Different from the Japanese-to-Chinese translation, we find that sharing BPE (Sennrich et al., 2015) tokens between Chinese and Japanese can produce better translation results. Besides, we can truncate the vocabulary size to accelerate model training. Table 2 shows the comparison of different vocabulary strategy. The number of BPE merge operations is 30k. In truncated version, the vocabulary is truncated to 40K BPE tokens.

Model	Share	Truncate	BLEU
ZH to JA	×	×	32.4
	×	√	32.6
	√	√	33.5

Table 2: Vocabulary strategy on Chinese-to-Japanese translation. The evaluation metric is 4-gram character BLEU score on the development set. In truncated version, the vocabulary is truncated to 40K BPE tokens.

3.2 Constructing Augmented Training Data

Following the work of Edunov et al. (2018), noise is added to the back-translation data. We delete a word with probability 10%, replace a word by a placeholder token with probability 10%, and swap words no further than three positions apart. Besides, we use bilingual data upsampling factor 4 to make the model pay more attention to the high-quality parallel data.

4 Model

We think the Transformer model is a strong model with excellent performance. So, we only take some small tricks on this model. In this section, we describe two different methods to enhance our model performance in this competition. All of them come from previous work (Sun et al., 2019; Shaw et al., 2018) and all of these methods help us to improve the baseline model. In the subsection, we will describe these methods briefly.

4.1 Bigger Transformer

In the work of (Sun et al., 2019), they proposed a method that increases the model capacity on the translation model and gets progress. Thus, we can think about if the model becomes wider, the performance may be better. We implement to increase the inner dimension of the feed-forward network in a big transformer model, from 4096 to 8192. Also, thinking about the overfitting problem, we increase the relu dropout value from 0.1 to 0.3.

4.2 Relative Position Representation

Recent empirical work shows that in the self-attention mechanism, it is better to use relative

System	Clean	Filtered	Re-Filtered	BT	Dev BLEU	Test BLEU
Baseline	√				20.0	22.0
	√	√			26.9	-
	√	√	√		28.6	-
	√	√	√	√	29.6	-
Bigger + RP*	√	√	√	√	30.3	34.0

Table 3: Results obtained by different data pre-processing methods and combinations. "Clean" denotes the data of Part A, "Filtered" denotes all training data filtered by the organizers, "Re-Filtered" denotes our re-filter method, "BT" is the abbreviation of back-translation, and "RP" means relative position. (* denotes our submitted system)

position (Shaw et al., 2018) to reflect the sequential relationship of words. In original ways, the Transformer only uses absolute position information in word embeddings. With the relative position feature, we compare the result and find it has better performance.

5 Submission to IWSLT 2020

5.1 Experiment

We compare the performance of our system on different data sets to show the effectiveness of data processing. In general, we adopt the default hyper-parameters of `transformer_relative_big` in tensor2tensor [3]. Except that we set the inner dimension of the feed-forward network to 8192, and set relu dropout to 0.3. We conduct our experiments on a machine with 8 Nvidia P40 GPUs. The model is updated 500K times in 9 days. Model parameters are saved every 1000 steps, and the last three checkpoints are averaged to obtain the final model. In decoding, we search the best decoding configuration on the released development set and fix the beam size as 6, alpha as 0.8. It is regretful that because of limited computational resources, we only trained a single model and didn't conduct model ensemble experiments.

As for post-processing, we process the decoding results by removing "UNK" token and Japanese kana characters from translated Chinese texts.

5.2 Japanese-to-Chinese Translation Results

Table 3 lists results obtained by using different training data. We use the official baseline system to test the effects of data processing. In the table, data size increases from the left columns to the right columns, and the performance is also improved. This shows the importance of extending training data in this task and validates the necessity of data pre-processing in boosting translation system accuracy. Also, the submitted system adopts a larger inner dimension and relative position, which shows the highest BLEU score in our systems.

[3] https://github.com/tensorflow/tensor2tensor

6 Conclusion

We participated in the Japanese-to-Chinese translation direction in the IWSLT 2020 open domain translation task. We focus on improving the Transformer baseline system by doing elaborate data pre-processing, and we manage to obtain significant improvements. Experiments also show that increasing model capacity is beneficial on large training data. Finally, our submission of Japanese-to-Chinese translation achieves the 2nd highest BLEU score among all the submissions.

Acknowledgments

We thank the anonymous reviewers for their valuable comments, Mengxia Zhai for assisting processing data and Ajay Nagesh for evaluating our translation result on the secret mixed-genre test dataset.

References

Ebrahim Ansari, Amittai Axelrod, Nguyen Bach, Ondrej Bojar, Roldano Cattoni, Fahim Dalvi, Nadir Durrani, Marcello Federico, Christian Federmann, Jiatao Gu, Fei Huang, Kevin Knight, Xutai Ma, Ajay Nagesh, Matteo Negri, Jan Niehues, Juan Pino, Elizabeth Salesky, Xing Shi, Sebastian Stüker, Marco Turchi, and Changhan Wang. 2020. Findings of the IWSLT 2020 Evaluation Campaign. In *Proceedings of the 17th International Conference on Spoken Language Translation (IWSLT 2020)*, Seattle, USA.

Chenhui Chu, Toshiaki Nakazawa, and Sadao Kurohashi. 2014. Constructing a chinese—japanese parallel corpus from wikipedia. In *LREC*, pages 642–647.

Sergey Edunov, Myle Ott, Michael Auli, and David

Grangier. 2018. Understanding back-translation at scale. *arXiv preprint arXiv:1808.09381*.

Armand Joulin, Edouard Grave, Piotr Bojanowski, Matthijs Douze, Hérve Jégou, and Tomas Mikolov. 2016a. Fasttext. zip: Compressing text classification models. *arXiv preprint arXiv:1612.03651*.

Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. 2016b. Bag of tricks for efficient text classification. *arXiv preprint arXiv:1607.01759*.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2015. Neural machine translation of rare words with subword units. *arXiv preprint arXiv:1508.07909*.

Peter Shaw, Jakob Uszkoreit, and Ashish Vaswani. 2018. Self-attention with relative position representations. *arXiv preprint arXiv:1803.02155*.

Meng Sun, Bojian Jiang, Hao Xiong, Zhongjun He, Hua Wu, and Haifeng Wang. 2019. Baidu neural machine translation systems for wmt19. In *Proceedings of the Fourth Conference on Machine Translation (Volume 2: Shared Task Papers, Day 1)*, pages 374–381. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, et al. 2016. Google's neural machine translation system: Bridging the gap between human and machine translation. *arXiv preprint arXiv:1609.08144*.