System Description of bjtu_nlp Neural Machine Translation System

Shaotong Li
Beijing Jiaotong University
15120415@bjtu.edu.cn

JinAn Xu
Beijing Jiaotong University
jaxu@bjtu.edu.cn

Yufeng Chen
Beijing Jiaotong University
chenyf@bjtu.edu.cn

Yujie Zhang
Beijing Jiaotong University
yjzhang@bjtu.edu.cn

Abstract

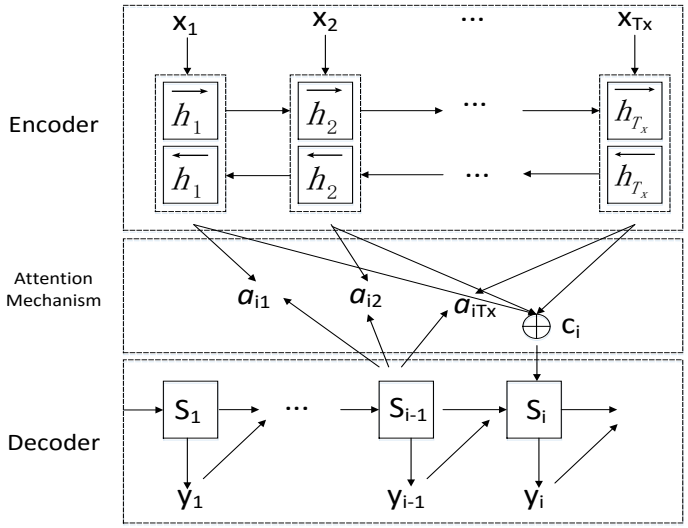
This paper presents our machine translation system that developed for the WAT2016 evaluation tasks of ja-en, ja-zh, en-ja, zh-ja, JPCja-en, JPCja-zh, JPCen-ja, JPCzh-ja. We build our system based on encoder–decoder framework by integrating recurrent neural network (RNN) and gate recurrent unit (GRU), and we also adopt an attention mechanism for solving the problem of information loss. Additionally, we propose a simple translation-specific approach to resolve the unknown word translation problem. Experimental results show that our system performs better than the baseline statistical machine translation (SMT) systems in each task. Moreover, it shows that our proposed approach of unknown word translation performs effectively improvement of translation results.

1 Introduction

Our system is constructed by using the framework of neural machine translation (NMT). NMT is a recently proposed approach to machine translation. Unlike the traditional SMT, the NMT aims at building a single neural network that can be jointly turned to maximize the translation performance (Kalchbrenner et al., 2013; Sutskever et al., 2014; Luong et al., 2014).

Most of the existing NMT models are built based on Encoder-Decoder framework (Sutskever et al., 2014; Luong et al., 2014). The encoder network encodes the source sentence into a vector, the decoder generates a target sentence. While early models encode the source sentence into a fixed-length vector. For instance, Bahdanau et al. advocate the attention mechanism to dynamically generate a context vector of the whole source sentence (Bahdanau et al., 2014) for improving the performance of the NMT. Recently, a large amount of research works focus on the attention mechanism (Cheng et al., 2015; Firat et al., 2016).

In this paper, we adopt RNN, GRU and attention mechanism to build an Encoder-Decoder network as our machine translation system. Figure 1 shows the framework of our NMT.



Proceedings of the 3rd Workshop on Asian Translation,
pages 104–110, Osaka, Japan, December 11-17 2016.

Figure 1: The framework of NMT. Where *x* and *y* denote embeddings of words in the source vocabulary and target vocabulary respectively, *h* means the hidden state of Encoder RNN, *s* is the hidden state of decode RNN, c_i is the context vector, *a* expresses the attention weight of each position.

Experiment results show that our system achieved significantly higher BLEU scores compared to the traditional SMT system.

2 System overview

Figure 2 shows the structure of our NMT system.

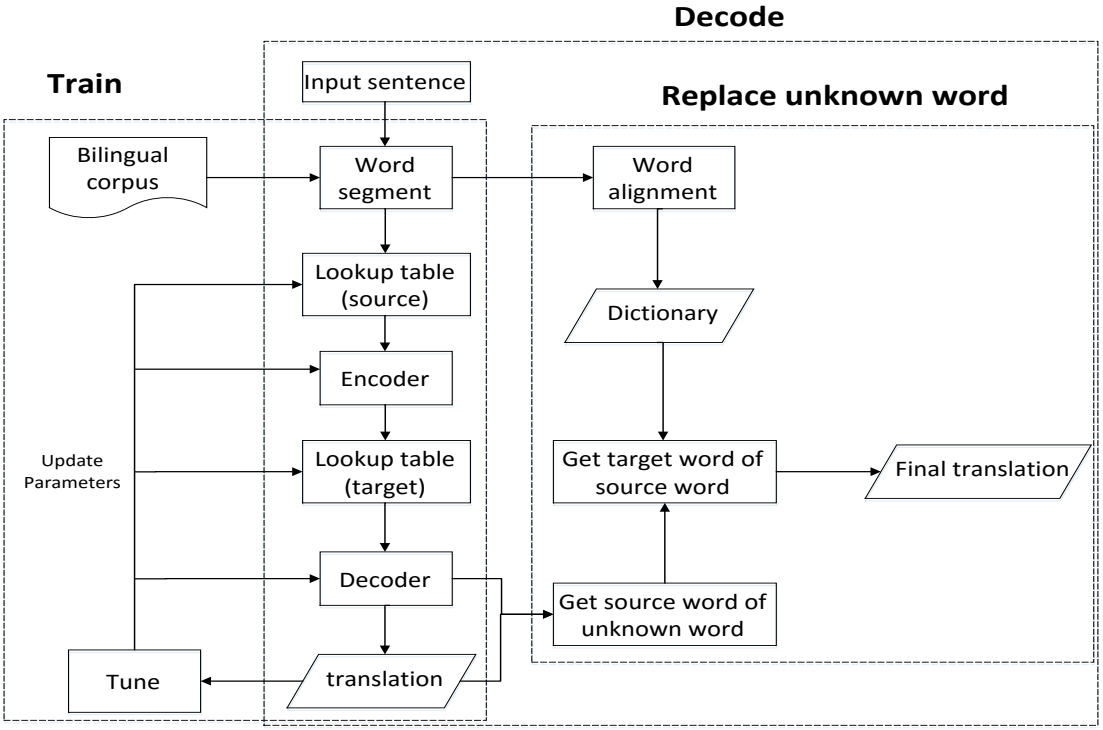


Figure 2: The structure of our system

Our system consists three parts: training part, decode part and the post-processing part of our proposed approach of unknown word processing.

2.1 Word segmentation

We use Stanford POS Tagger[1] and Juman[2] to do Chinese and Japanese segmentation processing, respectively. For English word segmentation, we use Moses tokenizer[3].

All these tools are the same as baseline systems tools.

2.2 Lookup table

For each word of source sentence， we obtain its embedding by using the source vocabulary, and for each target word of being predicted, we obtain its embedding with the target vocabulary. The source vocabulary and target vocabulary were regarded as part of the Encoder-Decoder network and the word embeddings will be tuned together with other parameters.

2.3 Encoder

In the encoder part, in order to make the annotation of each position of the source sequence, it consists two parts, both of the preceding words and the following words. We use a bidirectional RNN (BiRNN)

[1] http://nlp.stanford.edu/software/segmenter.shtml
[2] http://nlp.ist.i.kyoto-u.ac.jp/EN/index.php?JUMAN
[3] https://github.com/moses-smt/mosesdecoder/tree/RELEASE-2.1.1

to encode the source sentence (Schuster et al., 1997). We selecting GRU as the update function of hidden states of the BiRNN, which was proposed by Cho et al. (Cho et al., 2014) to make each recurrent unit to adaptively capture dependencies of different time scales.

2.4 Decoder

The decoder is constructed with another RNN, we use this RNN to predict each target word and finally generate an output sequence as the translated result sentence. We also select GRU as the update function of this RNN. We use a context vector which is dynamically generated by the attention mechanism (Bahdanau et al., 2014), as the input of the decode RNN.

2.5 Tune

After generating the output sequence, a softmax function is applied to calculate the cross-entropy as the cost which is used to compute grads of all parameters. We use the method of Adadelta (Zeiler et al., 2012) to tune the parameters.

2.6 Approach of Unknown Words translation problem

As the size of vocabulary of target language is limited owing to decoding complexity, there may be unknown words from the target vocabulary in the translation processing. This is a key point of existing NMTs.

In our system, we adopt a simple translation-specific approach to solve this problem. Firstly, we get a bilingual dictionary using GIZA++[4]. In decoding, each word, including unknown words, in the translation are matched with each word in the source, Secondly, we find the source word corresponding to unknown word with largest score in the decoder attention mechanism. For each unknown word, our approach can automatically select its corresponding word in the source sentence according to its matching scores. Then, we can use the translation of the corresponding source word to replace unknown word.

3 Evaluation

We participated in all tasks related to Chinese and Japanese and English.

3.1 Dataset

We use the given data of Asian Scientific Paper Excerpt Corpus (ASPEC)[5] and JPO Patent Corpus (JPC)[6] as show in table 1.

Corpus	Data Type	Number of sentences
ASPEC-JE	TRAIN	3000000
	DEV	1790
	TEST	1812
ASPEC-JC	TRAIN	672315
	DEV	2090
	TEST	2107
JPC-JE	TRAIN	1000000
	DEV	2000
	TEST	2000
JPC-JC	TRAIN	1000000
	DEV	2000
	TEST	2000

Table 1: Experimental dataset

[4] http://code.google.com/p/giza-pp/
[5] http://lotus.kuee.kyoto-u.ac.jp/ASPEC/
[6] http://lotus.kuee.kyoto-u.ac.jp/WAT/patent/index.html

For long sentence, we discarded all of the sentences which length with more than 50 words on both source and target side.

3.2 Training details

We defined hyper-parameters for each task as follows:

On ASPEC-JE corpus, the vocabulary size of English side is 40k while Japanese side is 30k. The number of hidden units is 1000 for both encoder and decoder. And the word embedding dimension is 600 for English side and 500 for Japanese side. For reducing training time and giving full play to the advantages of GPU, we choice 128 sentences as a batch to train together. The dropout rate (Srivastava et al., 2014) in the last layer of the network is set to 0.5 to avoid overfitting. For reducing searching space, we use beam-search algorithm (Tillmann et al., 2003) in the decoder, the beam size is set to 10.

On the other three corpuses, the hyper-parameters are the same, excepting the vocabulary size and word embedding dimension are different. They are set as fallows.

On ASPEC-JC corpus, the vocabulary size of Chinese side is 20k while Japanese side is 20k. And the word embedding dimension is 500 for Chinese side and 500 for Japanese side.

On JPC-JE corpus, the vocabulary size of English side is 30k while Japanese side is 30k. And the word embedding dimension is 500 for English side and 500 for Japanese side.

On JPC-JC corpus, the vocabulary size of Chinese side is 30k while Japanese side is 30k. And the word embedding dimension is 500 for Chinese side and 500 for Japanese side.

3.3 Evaluating results

We evaluated the performance of our two systems, one is the NMT system named as GRUSearch, the other is NMT system named as GRUSearch+UNKreplace, which adopted unknown word solution processing. For comparison, we also conducted evaluation experiments by using the three baseline systems provided by the organizers: Phrase-based SMT, Tree-to-String SMT, Hierarchical Phrase-based SMT.

For automatic evaluation, we use the standard BLEU and RIBES metrics. For human evalution, we use Pairwise Crowdsourcing Evaluation score provided by the organizers. The official evaluation results on ASPEC are shown in table 2, and the evaluation results on JPC are shown in table 3.

Task	System	BLEU	RIBES	HUMAN
en-ja	PB SMT	29.80	0.692	--
	HPB SMT	32.56	0.747	--
	T2S SMT	33.44	0.758	--
	GRUSearch	32.85	0.782	--
	GRUSearch+UNKreplace	**33.47**	**0.787**	**39.50**
Ja-en	PB SMT	18.45	0.645	--
	HPB SMT	18.72	0.651	--
	T2S SMT	**20.36**	0.678	--
	GRUSearch	17.67	0.679	--
	GRUSearch+UNKreplace	18.34	**0.690**	**19.25**
Zh-ja	PB SMT	35.16	0.766	--
	HPB SMT	35.91	0.799	--
	T2S SMT	37.07	0.820	--
	GRUSearch	37.83	0.837	--
	GRUSearch+UNKreplace	**39.25**	**0.846**	**49.00**
Ja-zh	PB SMT	27.96	0.789	--
	HPB SMT	27.71	0.809	--
	T2S SMT	28.65	0.808	--
	GRUSearch	28.21	0.817	--
	GRUSearch+UNKreplace	**30.57**	**0.830**	**46.25**

Table 2: Official automatic evaluation results on ASPEC

Task	System	BLEU	RIBES	HUMAN
JPCen-ja	PB SMT	34.26	0.728	--
	HPB SMT	36.61	0.779	--
	T2S SMT	37.65	0.797	--
	GRUSearch	40.00	0.833	--
	GRUSearch+UNKreplace	**41.16**	**0.840**	**39.50**
JPCja-en	PB SMT	30.80	0.730	--
	HPB SMT	32.23	0.763	--
	T2S SMT	34.40	0.793	--
	GRUSearch	38.13	0.836	--
	GRUSearch+UNKreplace	**41.62**	**0.852**	**41.63**
JPCzh-ja	PB SMT	38.51	0.779	--
	HPB SMT	39.52	0.802	--
	T2S SMT	39.45	0.810	--
	GRUSearch	38.24	0.820	--
	GRUSearch+UNKreplace	**39.72**	**0.831**	**32.25**
JPCja-zh	PB SMT	30.60	0.787	--
	HPB SMT	30.26	0.788	--
	T2S SMT	31.05	0.794	--
	GRUSearch	31.03	0.819	--
	GRUSearch+UNKreplace	**31.49**	**0.823**	**-1.00**

Table 3: Official automatic evaluation results on JPC

We also demonstrate the comparison results on BLEU and on RIBES in Figure 3 and Figure 4, separately.

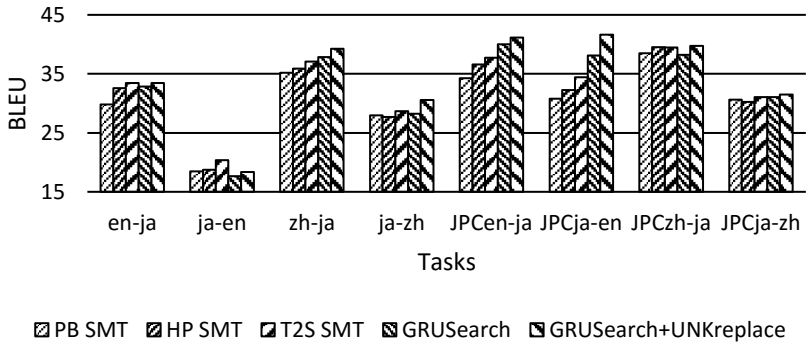


Figure 3: BLEU scores of all systems

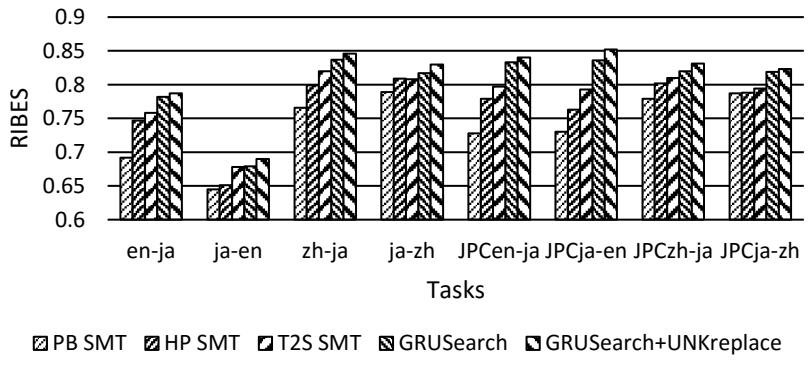


Figure 4: RIBES scores of all systems

As shown in the above tables and figures, our systems, both the GRUSearch+UNKreplace and GRUSearch outperformed the baseline systems in most tasks. In addition, our system with unknown word solution, GRUSearch+UNKreplace performed much better than the system without the unknown word solution, GRUSearch. It is proved that our unknown word translation approach is effective. Therefore, we submitted GRUSearch+UNKreplace to WAT2016 for human evaluation. And all the Pairwise scores of our tasks except JPCja-zh are much bigger than zero, which further proved that GRUSearch+UNKreplace performed better than baseline system.

Specifically, in the JPCja-en task, GRUSearch+UNKreplace achieved an improvement of 7.22 of BLEU score, compared with T2S SMT. GRUSearch+UNKreplace also achieved an improvement of 3.49 of BLEU, compared with GRUSearch. It means that the effectiveness of our unknown word resolution achieved good performance by the support of a better attention network, and a better dictionary, which obtained from higher quality of training data.

However, our model shows great difference in different tasks, in two tasks, our system performs even worse than the baseline systems. It is considered that we need do more works to find the best hyper-parameters of these tasks. The hyper-parameter optimization will be one of the most important tasks of our future work.

4 Conclusion

In this paper, we described our NMT system which used RNN and GRU, and we adopt the attention mechanism into the encoder–decoder network. We also presented a translation-specific approach to solve the unknown words translation problem. Experiment results show that our system performs good performance in most of the evaluation tasks.

However, there exists some space to improve the performance of our system: The solution for dealing with unknown words is still an open question; Hyper-parameter optimization is one of the most important tasks in NMT system. We also will try to integrate morphological features such as part-of-speech tags, syntactic dependency labels as input features into NMT systems, to improve model quality, aiming at further improvement of translation results.

Reference

Bahdanau, Dzmitry, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. Computer Science.

Cheng, Yong, et al. 2015. Agreement-based joint training for bidirectional attention-based neural machine translation. arXiv preprint arXiv:1512.04650.

Cho, Kyunghyun, et al. 2014. Learning phrase representations using RNN encoder-decoder for statistical machine translation. arXiv preprint arXiv:1406.1078.Firat, Orhan, Kyunghyun Cho, and Yoshua Bengio. 2016. Multi-way, multilingual neural machine translation with a shared attention mechanism. arXiv preprint arXiv:1601.01073.

Cho, Kyunghyun, et al. 2014. On the properties of neural machine translation: Encoder-decoder approaches. arXiv preprint arXiv:1409.1259.

Firat, Orhan, Kyunghyun Cho, and Yoshua Bengio. "Multi-way, multilingual neural machine translation with a shared attention mechanism." arXiv preprint arXiv:1601.01073 (2016).

Forcada, Mikel L., and Ramón P. Ñeco. 1997. Recursive hetero-associative memories for translation. In International Work-Conference on Artificial Neural Networks (pp. 453-462). Springer Berlin Heidelberg.

Kalchbrenner, Nal, and Phil Blunsom. 2013. Recurrent Continuous Translation Models. In EMNLP (Vol. 3, No. 39, p. 413).

Nakazawa, Toshiaki, et al. 2016. Overview of the 3rd Workshop on Asian Translation. Proceedings of the 3rd Workshop on Asian Translation (WAT2016).

Schuster, Mike, and Kuldip K. Paliwal. 1997. Bidirectional recurrent neural networks. IEEE Transactions on Signal Processing, 45(11), 2673-2681.

Sennrich, Rico, and Barry Haddow. 2016. Linguistic Input Features Improve Neural Machine Translation. arXiv preprint arXiv:1606.02892.

Srivastava, Nitish, et al. 2014. Dropout: a simple way to prevent neural networks from overfitting. Journal of Machine Learning Research, 15(1), 1929-1958.

Sutskever, Ilya, Oriol Vinyals, and Quoc V. Le. 2014. Sequence to sequence learning with neural networks. In Advances in neural information processing systems (pp. 3104-3112).

Tillmann, Christoph, and Hermann Ney. 2003. Word reordering and a dynamic programming beam search algorithm for statistical machine translation. Computational linguistics, 29(1), 97-133.

Toshiaki Nakazawa, Manabu Yaguchi, et al. 2016. ASPEC: Asian Scientific Paper Excerpt Corpus. Proceedings of the 10th Conference on International Language Resources and Evaluation (LREC2016).

Vinyals, Oriol, Suman V. Ravuri, and Daniel Povey. 2012. Revisiting recurrent neural networks for robust ASR. In 2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (pp. 4085-4088). IEEE.

Zeiler, Matthew D. 2012. ADADELTA: an adaptive learning rate method. arXiv preprint arXiv:1212.5701.