Sogou Neural Machine Translation Systems for WMT17

Yuguang Wang[*], Xiang Li[*†], Shanbo Cheng[*], Liyang Jiang[*], Jiajun Yang[*]
Wei Chen[*], Lin Shi[*], Yanfeng Wang[*], Hongtao Yang[*]
[*]Voice Interaction Technology Center, Sogou Inc., Beijing, China
[†]Key Laboratory of Intelligent Information Processing,
Institute of Computing Technology, Chinese Academy of Sciences
{wangyuguang, chengshanbo, chenweibj8871}@sogou-inc.com, lixiang@ict.ac.cn

Abstract

We describe the *Sogou* neural machine translation systems for the WMT 2017 Chinese↔English news translation tasks. Our systems are based on a multi-layer encoder-decoder architecture with attention mechanism. The best translation is obtained with ensemble and reranking techniques. We also propose an approach to improve the named entity translation problem. Our Chinese→English system achieved the highest cased BLEU among all 20 submitted systems, and our English→Chinese system ranked the third out of 16 submitted systems.[1]

1 Introduction

End-to-end neural machine translation (NMT) has recently been introduced as a promising paradigm with the potential to address many shortcomings of traditional statistical machine translation (SMT) systems, and has obtained state-of-the-art performance for several language pairs (Cho et al., 2014; Sutskever et al., 2014; Bahdanau et al., 2015; Sennrich et al., 2016a; Wu et al., 2016; Zhou et al., 2016). In this paper, we describe the *Sogou* NMT systems submissions for the WMT 2017 Chinese→English and English→Chinese translation tasks.

Overview of the systems can be described as follows: we implement a multi-layer attention-based encoder-decoder integrated with recent promising techniques in NMT, including that we use subword units based on byte pair encoding (BPE) rather than words as modelling units (Sennrich et al., 2016b) and layer normalization (Ba et al., 2016) to isolated layers. And we improve the performance using ensemble based four systems of the same network trained with different random seeds of parameter initialization.

In addition, we improve the performance further by reranking the *n*-best translation lists with some effective features, including the target-bidirectional models, target-to-source models, and *n*-gram language models. And we use another NMT model to translate the recognized person names for the Chinese→English task, in order to improve the performance of unknown named entity translation.

Our Chinese→English system achieved the highest cased BLEU among all 20 submitted systems, and our English→Chinese system ranked the third out of 16 submitted systems.

2 Neural Machine Translation

Our NMT model follows the common attentional encoder-decoder networks (Bahdanau et al., 2015). We implement a deep multi-layer Long Short Term Memory (LSTM) recurrent neural network for both the encoder and decoder. In our setup, the encoder has one bi-directional LSTM layer followed by two uni-directional LSTM layers. The decoder has three uni-directional LSTM layers. Similar to the conditional GRU used in DL4MT (Firat and Cho, 2016), we use conditional LSTM (cLSTM) for the top layer of decoder instead of standard LSTM. The encoder takes the model's input sequence as input and encodes it into a fixed-size context vector. We only use the bottom layer output of the decoder to obtain attentional context vector, which is used to predict next target word at the top layer of the decoder combining with the previous hidden state and the previously generated words.

We utilize layer normalization (Ba et al., 2016) to isolated LSTM layers, a method that adaptively learns to scale and shift the incoming activations of a neuron on a layer-by-layer basis at each time step. Layer normalization can stabilize the dynamics of hidden layers in the network and accelerate the convergence speed of deep neural networks.

[1] Automatic rankings are from http://matrix.statmt.org.

Proceedings of the Conference on Machine Translation (WMT), Volume 2: Shared Task Papers, pages 410–415

All the weight parameters are initialized uniformly in [-0.02, 0.02], except for the square matrix weight parameters are initialized by orthogonal initialization (Henaff et al., 2016). We use dropout for the models as suggested by (Zaremba et al., 2015). We clip the gradient norm to 1.0 (Pascanu et al., 2013). Our main NMT decoder with a beam size of 10 is used in all experiments. We validate the model every 10,000 mini-batches via BLEU on the newsdev2017 data. We use a mini-batch size of 128, a hidden layer size 1024, a word embedding layer size of 512, filter out sentence pairs whose length exceeds 40 words, and reshuffle the training data between epochs as we proceed.

We use Adam (Kingma and Ma, 2014) to train the model with a learning rate 0.0001. We use the multi-GPUs training framework via asynchronous SGD (Dean et al., 2012) and data parallelism (copies of the full model on each GPU). We train the model on a host server with eight NVIDIA Tesla M40 GPUs. We train four systems of the same network with different random seeds of parameters initialization, perform early stop for each system, and use a widely used, simple ensemble method (prediction averaging) based on the best model of each system in order to improve the performance.

3 Experiment Techniques

This section describes several techniques integrated in our NMT system.

3.1 Reranking

In order to get better translation result, we explore different NMT variant models and *n*-gram language models as features in the reranking framework.

Target right-to-left NMT Model: The quality of the prefixes of translation hypotheses is much higher than that of the suffixes (Liu et al., 2016). In order to alleviate this unbalanced output problem, a variant right-to-left (R2L) NMT mode is trained on the training data, but the target data is inversed. We inverse the *n*-best lists generated by the main NMT model and calculate the likelihood which represents the conditional probabilities of reversed translations given the source sentences.

Target-to-source NMT Model: Moreover, the translation may be inadequate and repeat or miss out some words (Tu et al., 2016). In order to cope with the inadequateness, we use the target-to-source (T2S) reconstruction model trained with the swapped source and target training data. Because we participated in both the Chinese→English and English→Chinese tasks, the T2S model of Chinese→English is just the main NMT model of English→Chinese, and vice-versa.

***N*-gram language models:** There exists a large amount of monolingual data for both Chinese and English. We train *n*-gram language models on each corpus and select the top *k*-best *n*-gram language models as reranking features based on perplexity (PPL) calculated on the newsdev2017 data. It is noted that we use character-level language models for English→Chinese task and word-level language models for Chinese→English. For English, the language model is trained on the "News Crawl: articles from 2016" provided by WMT 2016 has the lowest PPL, which is even much lower than the language model trained on English side of the training data.

We first generate an *n*-best lists with an ensemble model for a source sentence. Then we calculate the likelihood score with T2S and R2L models. We also use *n*-gram language models to compute PPL for the translation candidates. We treat each model score as an individual feature. We use *k*-batched MIRA (Cherry et al. 2012) to tune the weights for all the features. In order to get more diverse *n*-best lists, we also try to increase the beam size to further improve reranking.

3.2 NMT with Tagging Model

Translating rare words is hard for a conventional NMT model with a fixed relatively small vocabulary so that a single *unk* symbol is used to represent the large number of out-of-vocabulary (OOV) words.

Our proposed tagging model is similar to the placeholder mechanism (Crego et al., 2016), which aims at alleviating the rare words problem. When using tagging model to translate a sentence, we first use the pre-defined tags to replace the OOV words in the source sentence, then translate the source sentence with tags using the NMT model, and recover the tags in translation based on the attention weights and a bilingual translation dictionary finally.

The most significant difference between our tagging model and placeholder mechanism (Crego et al., 2016) is that we don't force beam search to generate tags, but only try to find exactly the same tag in the source side (if exists) when a tag is generated

in the translation, and choose the one with the highest alignment probability based on attention weights. Given this information, we can find the source side to which a target tag is aligned, and obtain the translation of source tag via a bilingual dictionary.

Zhang et al., (2016) incorporated bilingual translation dictionary by using the dictionary to generate training data, where the bilingual dictionary is an external resource. While our work is of higher efficiency and the bilingual dictionary is trained from our training data alone.

In this paper, we use our CRF-based named entity recognize (NER) tagger to obtain the tags (placeholders). We also build the bilingual translation dictionary from scratch based on the training data.

Bilingual Translation Dictionary: The bilingual dictionary is generated by the following steps:

- Data preparation. We label both source-side and target-side words in the training data with our NER tagger and combine multi-words labelled with named-entities tags to a single word with specific marks so that we can recover the word to the original form.
- Word alignment. The word alignment is generated by using GIZA++ (Och and Ney, 2003) given the above data.
- Translation pairs extraction. The translation pairs are extracted according to the word alignment. We only extract those pairs whose both source and target side words are person name tags (labeled by our NER tagger), and represent the tag as a *$TERM* symbol in this paper.

The bilingual translation dictionary can not only be used as a lookup dictionary for tagging model, but also as the training data for the neural person name translation model in Sec. 3.3.

3.3 Named Entity Translation

Due to most of rare words in news data are person named entities, we propose an approach to translate the person named entities with an external character-based encoder-decoder model trained on the extracted parallel person names from the training data for the Chinese→English task individually, in order to improve the performance of rare words translation.

For the person named entity translation model, the size of the Chinese vocabulary is 3000 characters, the size of the English vocabulary is 30 characters, the size of hidden layers is 512, the size of embedding is 256, the size of mini-batch is 128, the sentence pairs whose length exceeds 30 characters are filtered out, and the training data is re-shuffled between epochs as we proceed. We validate the model every 1000 mini-batches via BLEU on the sample validation data (100 Chinese-English person names pairs). We only train the model on a single GPU and perform early stop.

Because many person names can be translated by the model, we only focus on the remaining person names aligned to the *unk* symbols in the target side according to the attention weights. Given an input sentence, we first recognize the person named entities with our NER tagger, then generate BPE segmentation for the plain sentence, and mark each subword unit which is part of a person named entity with a single name-aware symbol finally. During decoding, the text with BPE marker is first translated by our NMT model. We mark the source tokens to which each target *unk* symbol is most aligned with the method of Luong et al. (2015). If the marked source token is also a part of person named, the original person name is recovered via the BPE marker. Then we replace the recovered person names with a single *$TERM* symbol. Finally, we translate the text with *$TERM* symbols and BPE marker again, and replace the target *$TERM* symbols with the translation of original person names generated by our neural person named entity translation model.

Chinese Person Name	Translation
史婧琳 (Shǐ jìng lín)	Shi Jinglin
安东 • 瓦伊诺 (Ān dōng • wǎ yī nuò)	Anton Vaino
法土拉 • 葛兰 (Fǎ tǔ lā • gě lán)	Fethullah Gulen

Table 1: Examples of neural person named entity translation.

Our proposed method is similar to Li et al. (2016), but we only use the extracted parallel person names from training data instead of Wikipedia data. Although our method brings no significant improvement on BLEU, we find that it is useful for human evaluation especially when the source data contains person names. The translation of person names in Table 1 seems like the transliteration of Chinese person names.

In addition, we also replace all the number named entities greater than 5000 of source sentences with a single number-aware symbol. Then

the number-aware symbols of translation are recovered to their original number named entities based the attention weights. Finally, the recovered number named entities are translated with human rules. By this mean, nearly most of number named entities can be translated correctly.

4 Experiments Settings and Results

4.1 Data Processing

The training data for the two translation tasks consists of 12 million sentences pairs, including all the CWMT 2017 training data and 3 million sentences selected from the UN corpus by calculating the PPL with an English language model trained on the News Crawl: articles from 2016. We used the official newsdev2017 as validation set for both Chinese→English and English→Chinese systems.

We first segmented the Chinese sentences with our Chinese word segmentation tool and tokenized English sentences with the scripts provided in Moses[2] (Koehn et al., 2007). Then we used BPE segmentation to process both source and target data. 300K subword symbols are used for the source side and 150K subword symbols are used for target side. For both Chinese→English and English→Chinese systems, the size of the source vocabulary and target vocabulary is 300K and 150K respectively. We created about 250K translation pairs for the bilingual dictionary described in Sec. 3.2.

4.2 Chinese→English Systems

Table 2 shows the Chinese→English translation results on validation set. We reported cased BLEU scores calculated with Moses' *multi-bleu.pl*[3] script. The baseline model is a conventional single-layer encoder-decoder model where we used a bi-LSTM layer for encoder and a cLSTM layer for decoder. Other settings are the same as our deep NMT model.

Our deep encoder-decoder model improves the baseline by 0.8 BLEU. In order to get more diverse models and better ensemble results, we trained four deep models independently with different random initializations. Then we selected the best model based on validation set from four systems for model ensemble. The ensemble result gives an additional improvement of 1.1 BLEU over the best single deep NMT system.

To evaluate the influence of person named entity translation on the performance of our NMT systems, we made an experiment on the newsdev2017 data. As a result, a little improvement by 0.1 BLEU is achieved. One reason for such little improvement is that the performance is calculated on word level, the translation of person name is regarded wrong even when there is only one letter difference. On the other hand, the amount of training data with *$TERM* symbols is insufficient, so that the model is incapable to learn as good as the plain data.

System	BLEU
baseline	19.4
+deep model	20.2
+ensemble (4 deep models)	21.3
+named entity translation	21.4
+reranking (1 R2L, 4 T2S)	21.7
+reranking (beam size 100)	22.4
+reranking (10 language models)	**22.9**

Table 2: Chinese→English BLEU results on development set. Submitted system is the last system.

Additionally, to recover the case information, a SMT-based recaser is trained on the English corpus with Moses toolkit[4]. And we also use a few simple uppercase rules, for example capitalizing the word at the beginning of a sentence.

According to the experiments in (Liu et al., 2016), a left-right/right-left reranking may also help increase diversity. Hereafter, we used one T2L model and four T2S models for reranking, resulting in a 0.3 BLEU improvement. Due to the limitation of beam search for NMT, we observed that most of *n*-best lists are very similar. By increasing the beam size from 10 to 100, we achieved another 0.7 BLEU improvement. We also evaluated the influence of *n*-gram language models for reranking. We trained several 5-gram language models and selected top ten best language models based on their PPL on validation set. We achieved another improvement by 0.5 BLEU. The last best system is our final submitted system.

[2] https://github.com/moses-smt/mosesdecoder/blob/master/scripts/tokenizer/tokenizer.perl
[3] https://github.com/moses-smt/mosesdecoder/blob/master/scripts/generic/multi-bleu.perl
[4] https://github.com/moses-smt/mosesdecoder/blob/master/scripts/recaser/train-recaser.perl

4.3 English→Chinese Systems

Table 3 shows the English→Chinese translation results on validation set. All results are evaluated by character-level BLEU. Similar to the Chinese→English systems, a shallow model and four deep models are trained independently. The deep model brings a 0.4 BLEU improvement over the shallow model baseline. The ensemble system improves by 1.1 BLEU over single best deep model. The NE replacement improves by 0.1 BLEU. We also trained one R2L model and four T2S models for reranking. These variant models improve the system by 0.8 BLEU. We observed a 0.2 BLEU improvement by increasing the beam size from 10 to 100. Finally, we trained five Chinese language models for reranking, including three word-level 5-gram language models and two character-level 5-gram language models, for re-scoring the *n*-best lists, resulting in a 0.5 BLEU improvement. The last system is our final submitted English→Chinese system.

For English→Chinese translation task, if a target *unk* symbol cannot be recovered by named entity tagging and translation model, we directly replace the target *unk* symbol with its aligned English word according to the attention weights.

System	BLEU
baseline	31.6
+deep model	32.0
+ensemble (4 deep model)	33.1
+name entity translation	33.2
+reranking (1 R2L, 4 T2S)	34.0
+reranking (beam size 100)	34.2
+reranking (5 language models)	**34.7**

Table 3: English→Chinese translation BLEU results on development set. Submitted system is the last system.

5 Conclusion

We present the *Sogou* NMT systems for WMT 2017 Chinese↔English news translation tasks. For both translation directions, our final systems are improved by 3.1~3.5 BLEU over baseline systems by using the following techniques: 1) a deep NMT model; 2) ensemble of diverse deep NMT models; 3) reranking *n*-best lists with NMT variant models and *n*-gram language models; 4) named entity tagging and translation model. Our submitted Chinese→English system achieved the highest cased BLEU among all 20 submitted systems, and our English→Chinese system ranked third out of 16 submitted system.

References

Cho, Kyunghyun, Van Merrienboer, Bart, Gulcehre, Caglar, Bahdanau, Dzmitry, Bougares, Fethi, Schwenk, Holger, and Bengio, Yoshua. 2014. Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation. In Proceedings of EMNLP, 2014.

Colin Cherry and Gorge Foster. Batch Tuning Strategies for Statistical Machine Translation, In Proceedings of NAACL, 2012.

Diederik Kingma and Jimmy Ba. Adam: A method for stochastic optimization. arXiv URL: https://arxiv.org/abs/1412.6980.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural Machine Translation by Jointly Learning to Align and Translate. In Proceedings of ICLR, 2015.

Koehn, P., Och, F. J., and Marcu, D. Statistical phrase-based translation. In Proceedings of NAACL, 2003.

Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. Sequence to sequence learning with neural networks. In Proceedings of NIPS, 2014.

Lei Jimmy Ba, Ryan Kiros, and Geoffrey E. Hinton. 2016. Layer normalization. CoRR 2016, arXiv URL: http://arxiv.org/abs/1607.06450.

Lemao Liu, Masao Utiyama, Andrew Finch, and Eiichiro Sumita. 2016. Agreement on Target-bidirectional Neural Machine Translation. In Proceedings of NAACL, 2016.

Jeffrey Dean, Greg S. Corrado, Rajat Monga, Kai Chen, Matthieu Devin, Quoc V. Le, Mark Z. Mao, MarcAurelio Ranzato, Andrew Senior, Paul Tucker, Ke Yang, Andrew Y. Ng. 2012. Large scale distributed deep networks. In Proceedings of NIPS, 2012.

Jiajun Zhang and Chengqing Zong. Bridging neural machine translation and bilingual dictionaries. URL: https://arxiv.org/abs/1610.07272.

Josep Crego, Jungi Kim, Guillaume Klein, Anabel Rebollo, Kathy Yang, Jean Senellart, Egor Akha-nov, Patrice Brunelle, Aurelien Coquard, Yongchao Deng, et al. 2016. Systran's pure neural machine translation systems. arXiv URL: https://arxiv.org/abs/1610.05540.

Mikael Henaff, Arthur Szlam, and Yann LeCun. Orthogonal RNNs and long-memory tasks. In Proceedings of ICML, 2016.

Orhan Firat and Kyunghyun Cho. Conditional gated recurrent unit with attention mechanism.

"github.com/nyu-dl/dl4mt-tutorial/blob/master/docs/cgru.pdf".

Razvan Pascanu, Tomas Mikolov, and Yoshua Bengio. On the difficulty of training recurrent neural networks. In Proceedings of ICML, 2013.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016a. Edinburgh Neural Machine Translation Systems for WMT 16. In Proceedings of the First Conference on Machine Translation, Berlin, Germany. Association for Computational Linguistics.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016b. Neural machine translation of rare words with subword units. In Proceedings of ACL, 2016.

Thang Luong, Ilya Sutskever, Quoc Le, Oriol Vinyals, and Wojciech Zaremba. Addressing the rare word problem in neural machine translation. In Proceedings of ACL, 2015.

Wojciech Zaremba, Ilya Sutskever, and Oriol Vinyals. 2015. Recurrent neural network regularization. In Proceedings of ICLR, 2015.

Xiaoqing Li, Jiajun Zhang and Chengqing Zong. 2016. Neural Name Translation Improves Neural Machine Translation. arXiv URL: https://arxiv.org/abs/1607.01856.

Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V. Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, Jeff Klingner, Apurva Shah, Melvin Johnson, Xiaobing Liu, Łukasz Kaiser, Stephan Gouws, Yoshikiyo Kato, Taku Kudo, Hideto Kazawa, Keith Stevens, George Kurian, Nishant Patil, Wei Wang, Cliff Young, Jason Smith, Jason Riesa, Alex Rudnick, Oriol Vinyals, Greg Corrado, Macduff Hughes, and Jeffrey Dean. 2016. Google's neural machine translation system: Bridging the gap between human and machine translation. arXiv URL: https://arxiv.org/abs/1609.08144.

Zhaopeng Tu, Yang Liu, Lifeng Shang, Xiaohua Liu, and Hang Li. 2016a. Neural machine translation with reconstruction. arXiv URL: https://arxiv.org/abs/1611.01874.

Zhou, Jie, Cao, Ying, Wang, Xuguang, Li, Peng, and Xu, Wei. 2016. Deep Recurrent Models with Fast-Forward Connections for Neural Machine Translation. arXiv URL: https://arxiv.org/abs/1606.04199.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondˇrej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: Open Source Toolkit for Statistical Machine Translation. In Proceedings of the ACL-2007 Demo and Poster Sessions, pages 177–180.