Sogou Neural Machine Translation Systems for WMT17

Yuguang Wang, Shanbo Cheng, Liyang Jiang, Jiajun Yang
Wei Chen, Muze Li, Lin Shi, Yanfeng Wang, Hongtao Yang
Voice Interaction Technology Center, Sogou Inc., Beijing, China
{wangyuguang, chengshanbo, jiangliyang, yangjiajun}@sogou-inc.com
{chenweibj8871, limuze, shilin, wangyanfeng, yanghongtao}@sogou-inc.com

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 attentional encoder-decoder model. The final result is rescored with target-bidirectional models, target-to-source models and ngram language models. We propose a neural person name translation model to improve the rare words translation problem. Our Chinese→English system achieved the highest 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 of addressing many shortcomings of traditional statistical machine translation 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 system submissions for the WMT 2017 Chinese to English and English to 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, include BPE subword segmentation (Sennrich et al., 2016b) and layer normalization (Ba et al., 2016). And we make an ensemble result based on the best models from four systems trained with different random seeds of parameter initialization.

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

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

2 Neural Machine Translation

Our model follows the common attentional encoder-decoder networks (Bahdanau et al., 2015). We implement a deep stacked Long Short Term Memory (LSTM) recurrent neural network for both the encoder and decoder. In our setup, the encoder has three layers, including one bi-directional layer and two uni-directional layers. And the decoder has three uni-directional layers. Similar to the conditional GRU in DL4MT (Firat and Cho, 2016), we use conditional LSTM for the top layer of decoder. We only use the bottom layer output of decoder to obtain recurrent attentional context, which is used only for the top layer.

We use the layer normalization (Ba et al., 2016), 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 the hidden layers in the network and accelerates the convergence speed of deep neural networks. We use mini-batch of size 128, filter out sentence pairs whose length exceeds 40 words, re-shuffle the training data between epochs as we proceed, hidden layers of size 1024, and word embeddings of size 512.

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

Our parameters are uniformly initialized in [−0.02, 0.02], except for the square matrix parameters are initialized by orthogonal initialization (Henaff et al., 2016). We train the models with Adam (Kingma and Ma, 2014). Additionally, we use dropout for our models as suggested by (Zaremba et al., 2015). We clip the gradient norm to 1.0 (Pascanu et al., 2013). We validate the model every 10,000 mini-batches via BLEU on the newsdev2017 data. We use the multi-GPU training framework via asynchronous SGD (Dean et al., 2012) and data parallelism (copies of the full model on each GPU). We training our systems on a host server with eight NVIDIA Tesla M40 GPUs. We train four systems with different random seeds of parameters initialization, perform early stop for single model, and use the best model of each system for the final ensemble results.

3 Experiment Techniques

This section describe several techniques integrated in our system.

3.1 Reranking

In order to get better translation result, we test different NMT variant models and ngram Language models in re-reranking n-best list.

Target right-to-left NMT Model: When the target words are decoded by the NMT system, the later words will depend on the previous words decisions in the beam search decoder. So the word decision at time step t is much harder than that of timestep t-1. (Liu et al., 2016). In order to alleviate this imbalance problem, a variant NMT model, which decodes the target words from right-to-left (r2l), is trained. The r2l model is used to re-rank the n-best list which produced by the main NMT model. The scores represents the conditional probabilities of the reversed translations given the source sentences.

Target-to-source NMT Model: Moreover, the translations may be inadequate: the translations may repeat or miss out some words (Tu et al., 2016). In order to cope with the inadequateness, we also test the target-to-source (t2s) model, which is trained with the source and target swapped. Because we participate the Chinese→English and English→Chinese tasks, the t2s model for Chinese→English is just the main NMT model of English→Chinese, and vice-versa.

Ngram Language Models: The news task provides a large amount of mono-lingual data for both Chinese and English. We train ngram Language models on each corpus and test PPL on the development set. Then we select top N best ngram LM as reranking features based on PPL calculated on development set. Here, we also add char-based Language models for English→Chinese systems. Among all the English LMs, the LM trained on "News Crawl: articles from 2016" has lowest PPL, which is much lower than English LM trained on English side of the parallel corpus.

We first produce one n-best list with an ensemble of serval models. Then we do force decoding with target right-to-left, target-to-source NMT models. We also use language models to score the translation. We treat each models scores as an individual feature. We use k-batched MIRA (Cherry et al. 2012) to tune weights for all the features. In order to get more diverse n-best list, we also try to increase the size of beam from 10 to 100 for reranking.

3.2 NMT with Tagging Model

Conventional NMT systems are incapable of translating rare words, because they have a fixed relatively small vocabulary which forces them to use a single UNK symbol to represent the large number of out-of-vocabulary (OOV) words.

Our tagging model is similar to the placeholder mechanism (Crego et al., 2016), which aims at alleviating the rare word problem.

When using tagging model (or placeholder mechanism) to translate a sentence, we firstly use pre-defined tags to replace the rare words in the source sentence, then translate the replaced sentence with NMT model; finally, we recover the tag translation based on attention weights and a bilingual translation dictionary.

The most significant difference between our tagging model and placeholder mechanism is that, we don't force beam search to generate tags, but only when a tag is generated in the translation, we try to find exactly the same tag in the source side (if exists), and choose the one with the highest alignment probability based on attention weights. Given this information, we can find the source side of a translated tag, and recover the translation via bilingual dictionary.

Zhang et al., (2016) incorporates 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 NMT training data.

Bilingual Translation Dictionary: We generate the bilingual dictionary by the following steps:

- Data preparation. Using NER tagger to la-bel both source-side and target-side words in training data, and combining multi-words named-entities to single words with specific marks so that when there are no ambiguities when recover them to the original form.
- Word alignment. Using GIZA++ (Och et al., 2012) to train word alignment given the above data.
- Translation pair extraction. Extracting translation pairs according to word alignment. We only extracted those pairs where both source and target side words are of “person” tag (labeled by NER tagger), and defined the tag as “*$TERM*” 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 Section 3.3.

3.3 Neural Person Name Translation

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

We limit the source and target character vocabularies to 3000 and 30 respectively. We use mini-batch of size 128, filter out sentence pairs whose length exceeds 30 characters, re-shuffle the training data between epochs as we proceed, hidden layers of size 512, and character embeddings of size 256. We validate the model every 1000 mini-batches via BLEU on the sample validation data (100 person names). We only train the model on a single GPU and perform early stop.

Because some person names can be translated by the model, we only focus on the remaining person names whose aligned translation in the target side are the UNK symbols. Given an input sentence, we first recognize the person names with the NER tagger used in Section 3.2, and then make BPE subword segmentation for the data, at last mark each subword which is part of a person name. We translate the reformatted text with BPE marker using our NMT system. We mark the source tokens aligned to the UNK symbol in the translation with the method of Luong et al. (2015). If the source token aligned to UNK symbol is marked as a person name, the corresponding person name will be recovered via the BPE subword marker. Then we replace the rare person names with the single $TERM symbol. At last, we translate the text with $TERM symbols again, and replace the $TERM symbol of translation with the corresponding translation of person name generated by our neural person name translator.

To evaluate the influence of person name translation on the performance of our NMT systems, we make an experiment on the newsdev2017 data with our proposed person name translation method. As a result, we find a little improvement by 0.1 BLEU. One hand is that the performance is calculated in word level, the person name translation is regarded wrong even when there is only one letter difference. Our neural person name translation model achieves a surprising good result. The translation of person names in Table 1 seems like the pronunciation of Chinese names. Another hand is that the number of training data with $TERM symbol is insufficient, so the model is incapable to learn as good as the plain data.

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, we find the person name translation is very useful for human evaluation.

Chinese Name	Translation Name
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 name translation.

3.4 Post Processing

In addition to the NMT model and reranking systems, we still incorporate simple rules as supplement:

- Digit translation. We make classes for digits and substitute them in the end based on the attention. Digits are translated with rules.
- In English to Chinese translation, if a UNK cannot be proper treated by BPE and tag model, we directly copy the source-side word to translation according to attention weights.
- English recaser: To recover the case information, we trained an SMT-based recaser on the English corpus with Moses SMT toolkit[2]. And we also used a few simple uppercase rules, for example capitalizing words at the beginning of a sentences.

4 Experiments Settings and Results

4.1 Data Processing

The training data for both directions has 12 million sentences pairs, which consist of all the CWMT data and three million sentences selected from UN corpus. We first score each UN sentence with an English language model trained on News Crawl: articles from 2016, and then select sentences with lower PPL. We use the official newsdev2017 (2002 sentences) as validation set for both Chinese→English and English→Chinese systems.

We first segment the Chinese text with our Chinese word segmentation tool and tokenize English text with the Moses tokenizer[3]. Then we use BPE subword segmentation to process both source and target corpora. 300K BPE symbols are used for the source side and 150K BPE symbols for target side. For both Chinese→English and English→Chinese system, the size of the source vocabulary and target vocabulary is 300K and 150K. We created about 250,000 translation pairs for the bilingual dictionary described in Section 3.2.

4.2 Chinese→English Systems

Table 2 shows the Chinese→English translation results on validation set. All the results are cased BLEU evaluate by multi-bleu.perl script in moses[4]. The baseline model is a conventional single-layer encoder-decoder model where we use a bi-LSTM layer for encoder and a cLSTM layer for decoder. Other settings are the same as our deep NMT model.

system	dev
baseline	19.4
+deep model	20.2
+ensemble(4 deep models)	21.3
+NE replacement	21.4
+reranking(1 r2l, 4 t2s)	21.7
+reranking(beam:10→100)	22.4
+reranking(10 ngram LMs)	**22.9**

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

Our deep encoder-decoder model improves the baseline by 0.8 BLEU. In order to get more diverse models and better ensemble results, we train four deep models independently with different random initializations. Then we select the best model based on validation set from four systems for model ensembles. The ensemble result gives an improvement of 1.1 BLEU over best single system. The tag model and the neural person name translation also improve the system by 0.1 BLEU. The reason the tagging model brings out little improvement to BLEU score are two folds: one is the scale of our training data for tagging model is relatively small, which might lead to under-fitting problem; the other is that BPE can reduce the OOV rate significantly. Our tagging model and Neural Person Name translation model are used to cover the UNKs which cannot be proper treated by BPE.

According to experiments in (Liu et al., 2016), a left-right/right-left reranking may also help increase diversity to ensembles. Hereafter, we add a right-to-left and four target-to-source model for reranking, which improve the system by 0.3 BLEU. Due to the limitation of beam search for NMT, we found most translation in the n-best list are very similar. By increasing the width of beam from 10 to 100, we achieve another improvement of 0.7 BLEU. We also test ngram language models for reranking. We trained several ngram language models and select top ten best language models based on their PPL on validation set. The addition of ten ngram language models reranking for improved the

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

[3] https://github.com/moses-smt/mosesdecoder/blob/master/scripts/tokenizer/tokenizer.perl

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

system by 0.5 BLEU. The last system is our final submitted system.

4.3 English→Chinese Systems

Table 3 shows the English→Chinese translation results on validation set. All results are char-based BLEU. As with the Chinese→English system, a shallow model and four deep models are trained independently. The deep model improves 0.4 BLEU over the shallow model baseline. The ensemble system improves 1.1 BLEU over single best system. The NE replacement improves by 0.1 BLEU. We also train a right-to-left (r2l) system and take the four Chinese→English NMT model as the t2s models. These variant models improve the system by 0.8 BLEU. The size of n-best list is increased from 10 to 100, and we observe an increase of 0.2 BLEU. Finally we trained five Chinese language models, including three word-based ngram language models and two char-based ngram language models, for re-scoring the n-best list. The language models get 0.5 BLEU improvements. The last system is our final submitted English→Chinese system.

System	dev
baseline	31.6
+deep model	32.0
+Ensemble(4 deep model)	33.1
+NE replacement	33.2
+reranking(1 r2l, 4 t2s)	34.0
+reranking(beam:10→100)	34.2
+reranking(3 word,2 char ngram LMs)	**34.7**

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

5 Conclusion

We present the *Sogou* Chinese↔English systems for WMT 2017 shared news translation task. For both translation directions, our final systems are improved by 3.1~3.5 BLEU over baseline systems using the following techniques 1) deep encoder-decoder model 2) ensembles of diverse NMT models 3) reranking n-best list with NMT variant models and ngram language models 4) tagging model and neural person name translation model. Our Chinese→English system achieved the highest BLEU among all 20 submitted systems, and our English→Chinese system ranked third out of 16 submitted system in BLEU.

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 Proc. of EMNLP, 2014.

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

Diederik Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. arXiv URL: https://arxiv.org/abs/1412.6980.

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

Franz Josef Och, Hermann Ney. 2003. "A Systematic Comparison of Various Statistical Alignment Models", Computational Linguistics, volume 29, number 1, pp. 19-51 March 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 NAACL HLT 16, San Diego, CA.

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 NIPS. 2012.

Jiajun Zhang and Chengqing Zong. 2016. 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. 2016. Orthogonal RNNs and long-memory tasks. In ICML 2016.

Orhan Firat and Kyunghyun Cho. 2016. Conditional gated recurrent unit with attention mechanism.

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

Razvan Pascanu, Tomas Mikolov, and Yoshua Ben- gio. 2013. On the difficulty of training recurrent neural networks. In Proceedings of ICML 2013, pages 1310–1318, Atlanta, GA, USA.

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. 2015. Addressing the rare word problem in neural machine translation. In Proceedings of ACL, pages 11–19.

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