Russian-English Bidirectional Machine Translation System

Ariel Xv [1]
Beihang University, Beijing 100191, China
zf1906251@buaa.edu.cn
Wenhan Chao [2]
Beihang University, Beijing 100191, China
chaowenhan@buaa.edu.cn

Abstract

This review depicts our submission to the WMT20 shared news translation task. WMT is the conference to assess the level of machine translation capabilities of organizations in the word. We participated in one language pair and two language directions, from Russian to English and from English to Russian. We used official training data, 102 million parallel corpora and 10 million monolingual corpora. Our baseline systems are Transformer models trained with the Sockeye sequence modeling toolkit, supplemented by bi-text data filtering schemes, back-translations, reordering and other related processing methods. The BLEU value of our translation result from Russian to English is 35.7, ranking 5th, while from English to Russian is 39.8, ranking 2th.

1 Introduction

We participated in WMT20 shared news translation task by building neural translation systems for one language pair and two language directions, from English to Russian and from Russian to English. Our systems are based on the framework of the Transformer neural machine translation model, using many techniques and approaches, including the use of BPE subword segmentation for open-vocabulary translation with a fixed vocabulary, large-scale back-translation,and model ensembling.

Neural machine translation(Bahdanau et al., 2014) has emerged as the most promising machine translation approach in recent years, showing superior performance on public benchmarks. The proposed attention mechanism brought a new revolution in the neural machine translation in most cases, making the overall effect of translation much better than before. Then, the Transformer(Vaswani et al., 2017) that makes full use of the attention mechanism demonstrated outstanding performance and effectiveness. Up to now, most of work uses the structure of Transformer, and its superiority has been widely recognized.

Since the beginning of machine translation research, the translation between Russian and English has been extensively developed.As early as 1954, Georgetown University in the United States under the IBM company completed the English-Russian machine translation experiment with the IBM-701 computer, which opened the prelude of machine translation research. During the period, there are three core technologies, rule-based machine translation, statistical machine translation and neural machine translation. However, as the application field of machine translation became more and more complex, the limitations of various technologies started to become obvious. Due to more application scenarios and higher requirements for accuracy, model optimization problems appeared.

The translation between Russian and English is extremely difficult because their linguistic features are distinguished and the lexical composition and grammatical structure of Russian are more complicated than those of English. In the early period,statistical machine translations were hoped to be implemented through phrase-based methods(Marcu and Wong, 2002) and related techniques for language models and translation models. These methods have solved the Russian-English translation problems to a certain extent. Yet, at the same time, there exists translation problems that are high time cost and poor translation effect.

Since then, the emergence of neural machine translation has brought new developments to Russian-English machine translation. The basic modeling framework for neural machine translation is an end-to-end sequence generation model, a framework and method for transforming input sequences into output sequences. There are two

Proceedings of the 5th Conference on Machine Translation (WMT), pages 320–325

points in the core part. One is to represent the input sequence through the encoder, and the other is to obtain the output sequence through the decoder. In addition, for machine translation, neural machine translation not only includes encoding and decoding, but also uses RNN(Sutskever et al., 2014) or other methods to encode sentence pairs. It also introduces an additional mechanism, the attention mechanism(Luong et al., 2015), to help us to convert sequences. These innovations lead to an increase in translation performance in comparison to earlier models. Later, Transformer appeared, which greatly enhances the neural machine translation performance.

This paper is based on Transformer, a neural machine translation network structure, to develop a two-way evaluation task between Russian and English. Taking into account the language characteristics of Russian and English, we have done appropriate operations in data preprocessing, including removing duplicates, deleting unreasonable sentence pairs, lowercase and Latinization operations, and judging sentence alignment problems, removing the parallel corpus with problems. The filtered parallel corpus is then sent to the model for training and the training results are tested.After getting the trained model, we start to consider using the back-translation operation to augment the data, continuing to filter the generated artificial corpus, and put it into the model training together with the original parallel corpus.

Finally, ensemble(Dietterich, 2000), average and rerank(Shen et al., 2004) operations are implemented on different models to improve the overall performance of the translation system.

2 Background

Neural network machine translation is based on a sequence-to-sequence overall structure consisting of an encoder and a decoder. The encoder converts the source language sentence into an intermediate sequence result, and the decoder converts the intermediate sequence result into a target language sentence. There is also the Attention mechanism to help make the results perform better. In the construction of the overall translation system, we used a lot of excellent methods proposed earlier in the literature.

The basic model used here is Transformer, introduced by(Vaswani et al., 2017) . The transformer is an attention-based structure proposed to deal with tasks that require sequence models, such as machine translation. Traditional neural machine translation mostly uses RNN or CNN as the model base of encoder-decoder, and Google's latest Attention-based Transformer model abandons the inherent formula and does not use any CNN or RNN structure. The model works in high-level parallel process, so training speed is also relatively fast while improving translation performance. But it is still computationally expensive.

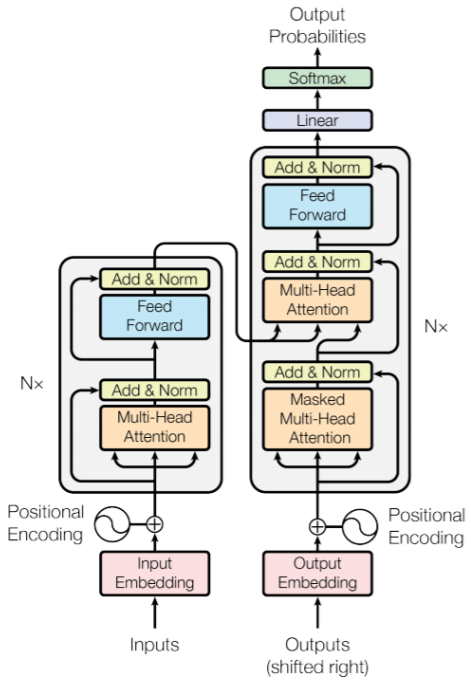

Figure 1: Transformer Structure.

The structure of Transformer is shown in Figure 1. The model is divided into two parts: the encoder and the decoder. The encoder is stacked by six identical layers, each with two more sub-layers. The first sub-layer is a long self-attention mechanism, and the second sub-layer is a simple fully connected feed forward network. A residual connection is added outside the two layers, and then layer normalization is performed. The output dimensions of all sub-layers and embedding layers of the model are d_{models}; the decoder also stacks six identical layers. However, in addition to the two layers in the encoder, the decoder also adds a third sub-layer, as shown in the figure which also uses the residual and layer normalization.

3 Data

We use all available bitext data which provided by WMT for the Russian-English language pair. For the monolingual data we use English and Russian Newscrawl as well as a filtered part of Commoncrawl in Russian. We choose to use Russian Commoncrawl to augment our monolingual data due to the relatively small size of Russian Newscrawl compared to English.

3.1 Data preprocessing

For the Russian-English language pair, we applied a series of preprocessing steps using scripts available in the Moses decoder(Koehn et al., 2007):

- replacing unicode punctuation,
- removing non-printing characters,
- normalizing punctuation,
- tokenization.

Also, we use joint byte pair encodings(BPE) with 32K split operations for subword segmentation(Sennrich et al., 2015) for each language.

3.2 Data Filtering

The large datasets which were crawled from the web would naturally be very noisy. And if they are used in their original and raw format, it may reduce the overall performance of the system. Clearning up these datasets is an important step to achieve good performance on any downstream tasks.

We applied two types of filters for data filtering: one is rule-based heuristics and another are filters based on language identification(Joulin et al., 2016).

For the Russian-English bitext data we used some data preprocessing methods to filter out them including:

- removing the bitext sentence pairs with a fixed length ratio above a certain threshold: for all the datasets we used a threshold of 3.
- removing sentence pairs with too short sentences: for all the sentences pairs we required a minimum number of five words.
- removing sentence pairs with too long sentences: we restricted all data to a maximum length of 100 words.

	En-Ru
No filter	112294588
+ length filter	102154821
+ langid filter	90826580

Table 1: Number of sentences pairs for different filtering schemes.

	En	**Ru**
Newscrawl	33600797	22348032
+ langid filter	32538613	20989583

Table 2: Number of sentences pairs for different filtering schemes.

Through observing the parallel data, we found that there is a surprisingly large amount of text segments in a wrong language in all provided parallel training data. So after some random inspection of the data, it is necessary to apply off-the-shelf language identifiers to the data for removing additional erroneous text from the training data. We apply language identification filtering called langid(Lui et al., 2012)which can classify each sentence in the parallel corpus.

So we can keep only sentence pairs with correct languages on both sides. At last, we filter out about 15% of the original parallel data. See Table 1 for details on the bitext dataset sizes.

For the monolingual English and Russian Newscrawl data we also apply langid filtering. As the monolingual Newscrawl data for Russian is relatively smaller than that of English, we have to augment the Newscrawl data for Russian with monolingual data from commoncrawl corpus.But there is a problem that the quality of commoncrawl corpus is very poor but is also noisy.

4 Experiment

For this evaluation task, we first start from the data preprocessing, through data expansion operations to obtain the data that needs to be trained, and then input the Transformer model for training. We test the training results and finally ensemble results according to the model generated by different strategies, average and rerank operations,for the best results. Next, the specific experiment content will be presented separately. The overall project process is showed in Figure 2.

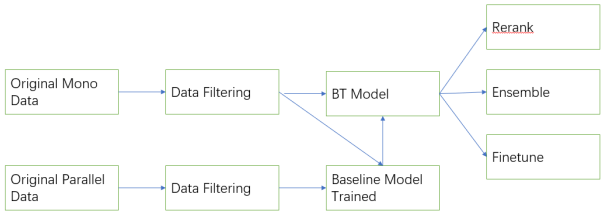

Figure 2: Project Process.

4.1 Base System

Our base system is based on the Transformer architecture (Vaswani et al., 2017) as implemented in Sockeye(Hieber et al., 2017). Due to the time cost and hardware cost of the evaluation task, we choose the basic version of Transformer. The encoder and decoder respectively have 6 sub-layers and the multi-head attention mechanism has 8 heads. The word embedding vector size is 512. We trained all our models using MXNET, which is the deep learning library that Amazon chose. The parameters setting of our models are listed in Table 3.

After the above processing, we use the parallel corpus which provided by the task organizers and direct it into the model for training and testing. The results of the base model can be used to generate reverse translation data to augment the corpus and continue training. The purpose is to maintain the generalization ability and robustness of the model to the greatest extent, and to provide reference for other model training results.

4.2 Large-scale Back-translation

Back-translation(Edunov et al., 2018) is an effective method to improve neural machine translation with monolingual data. It can incorporate monolingual data into a translation system. Firstly,we trained a baseline model that is used to translate monolingual target data into additional synthetic parallel data. This data is used in conjunction with original bitext data the desired source-to-target system.

In this work, due to the training time cost limitation we respectively only selected 10 million Russian and English sentences from the official monolingual corpus for back translation operations. We used back-translations obtained by beam-search(Edunov et al., 2018) from an ensemble of two target-to-source models. We adopt the method to tune the amount of bitext and pseudo-parallel corpora the model is trained on. We found that a ratio of 1:1 synthetic to bitext data can perform the best.

4.3 Fine-tuning

Fine-tuning is a common and effective method to improve machine translation quality especially for a downstream task. When we complete training on the original bitext and pseudo-parallel data, we train an special epoch on a smaller domain-specific data. It can make the model more sensitive to specific domain scenation and then get better results. Here, we select a corpus with much similarity to the test set from the training set to fine-tune the trained model. The similarity scores between the test corpus and the training corpus are sorted and ranked. Then the parallel sentence pairs with higher scores are found and the corpus is extracted as a fine-tuning corpus. In this way, about 5,000 pieces of data are obtained and this part of the corpus is input into the previously trained model to obtain the result of fine-tuning the model, so that it can perform better on the test set.

4.4 Model Reranking

N-best reranking is a method of improving translation quality by scoring and selecting a candidate hypothesis from a list of n-best hypotheses generated by a trained model. Extracting only one of the highest-scoring statements from the translation results of the model as an output is not necessarily the best result. So this strategy can be used to extract the best three from each translation model result as a candidate set. Then use some rules to rerank and get the best one as the output result. The translated content thus obtained is the comprehensive output of multiple results of each model. The rules used here include weighted summation of beam search score and the language model scores. The first one is based on the beam score returned during decoding, but different models have different performances, so it is difficult to sort under a uniform metric. So we introduced different weights for different models. Using beam score weight as the final score for each translation result, the final result was obtained by screening. The second one gives scores of the generated translations using the pre-trained language model. They are judged from the linguistics itself and the sentences with the highest scores are selected. The final result is an output that combines the highest scores of the two methods described above.

The above models also had different batch sizes,

Parameters	Transformer
optimizer	adam
max-num-checkpoint-not-improved	16
num_words	50000:50000
optimized-metric	perplexity
max-seq-len	100:100
loss	cross-entropy

Table 3: The parameters setting of Transformer are implemented by Sockeye .

comparison of the number of graphics cards and vocabulary sizes in the training process. We extracted them for the optimal results. Finally, the output is simply post-processed. In order to comply with common practice in natural language processing. However, due to the limitations of time and hardware resources, not every experiment has been refined and detailed totally, so there is still improvement of results in the future.

4.5 Ensemble Model

Ensemble is a method that combines the results of multiple models. The purpose of this is to complement the advantages of different models, make up for the problems that fall into the local optimum and get the results of the machine translation model with better comprehensive effects. For the sake of simplicity, only different initialization random seed parameters are set for the same model. So training of multiple models is performed, generally two or three models, and finally the results of all models are subjected to ensemble operation. By composing and complementing multiple models, we obtain the comprehensive optimal results of data translation.

5 Results

Results and ablations from Russian to English are shown in Table 4, from English and Russian are shown in Table 5. We report case-sensitive SacreBLEU scores using SacreBLEU(Post, 2018). We report all the case-sensitive BLEU(Koehn et al., 2007) score of our submitted system on this year's test set.

5.1 Russian To English

From Russian to English, we can see that langid filtering and ensembling improve our baseline performance on this year's test set by about 0.7 BLEU. This is perhaps due to the addition of higher quality bitext data and improved data filtering techniques. The addition of back-translated(BT) data

Type of Text	Pair	Bleu	Improve
base-re	RU-EN	33.1	0
filter-re	RU-EN	34.2	+1.1
ensemble-re	RU-EN	36.6	+2.4
fintune-re	**RU-EN**	**39.1**	+2.5
rerank-re	RU-EN	38.2	-1.1

Table 4: Russian-English Experiment Result.

Type of Text	Pair	Bleu	Improve
base-re	EN-RU	23.1	0
filter-re	EN-RU	24.2	+1.1
ensemble-re	EN-RU	24.5	+0.3
fintune-re	**EN-RU**	**24.8**	**+0.3**
rerank-re	EN-RU	24.6	-0.2

Table 5: English-Russian Experiment Result.

improves single model performance by about 0.3 BLEU, combining this with fine-tuning and ensembling gives us a total of 3 BLEU. We composed two models which have different random seeds and then re-trained on the fine-tuning corpus. Finally, applying reranking on top of these strong ensembled systems gives another 1.4 BLEU.

5.2 English To Russian

From English to Russian, we observe similar trends to Russian to English, with langid filtering and ensembling improving performance of a baseline system by 1.6 BLEU. Back-translatoin adds 1.5 BLEU, again mostly likely due to the lower quality bitext data available. Also we composed two models which have different random seeds and then re-trained on the fine-tuning corpus. Fine-tuning, ensembling, and reranking add almost 3 BLEU, with reranking contributing 1.2 BLEU.

6 Conclusions

This paper describes our submission to the WMT20 news translation task. In the evaluation task, we es-

tablished a Russian-English bidirectional machine translation system based on Transformer. For translations between Russian and English, we use the same strategy of filtering bitext data, performing beam-search back-translation on monolingual data. Then we train strong individual models on a combination of this data. Each of these models is fine-tuned and ensembled into a final system that is used for decoding with model reranking. In the final list, we got 2th in Ru-En, and 5th in En-Ru. Good results have been obtained in limited time and hardware resources, which is also in line with the industry's demands for service construction. In the whole experiment process, we also gained a lot of experience in data processing and experimental design, which will be of great help in later research and study. We will continue to improve the previous experiments, strive to get better results, and see what rankings can eventually be achieved, in preparation for the next year.

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

Thomas G Dietterich. 2000. Ensemble methods in machine learning. In *International workshop on multiple classifier systems*, pages 1–15. Springer.

Sergey Edunov, Myle Ott, Michael Auli, and David Grangier. 2018. Understanding back-translation at scale. *arXiv preprint arXiv:1808.09381*.

Felix Hieber, Tobias Domhan, Michael Denkowski, David Vilar, Artem Sokolov, Ann Clifton, and Matt Post. 2017. Sockeye: A toolkit for neural machine translation. *arXiv preprint arXiv:1712.05690*.

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

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, et al. 2007. Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th annual meeting of the ACL on interactive poster and demonstration sessions*, pages 177–180. Association for Computational Linguistics.

Minh-Thang Luong, Hieu Pham, and Christopher D Manning. 2015. Effective approaches to attention-based neural machine translation. *arXiv preprint arXiv:1508.04025*.

Daniel Marcu and Daniel Wong. 2002. A phrase-based, joint probability model for statistical machine translation. In *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing (EMNLP 2002)*, pages 133–139.

Matt Post. 2018. A call for clarity in reporting bleu scores. *arXiv preprint arXiv:1804.08771*.

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

Libin Shen, Anoop Sarkar, and Franz Josef Och. 2004. Discriminative reranking for machine translation. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics: HLT-NAACL 2004*, pages 177–184.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pages 3104–3112.

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.