Vietnamese-English Translation with Transformer and Back Translation in VLSP 2020 Machine Translation Shared Task

LE Duc Cuong **NGUYEN Thi Thu Trang***
School of Information and Communication Technology
Hanoi University of Science and Technology
cuongad1999@gmail.com *trangntt@soict.hust.edu.vn*

Abstract

Transformers have been proven to be more effective for machine translation and many NLP tasks. However, those networks may not work well to low-resource translation tasks, such as the one for the English-Vietnamese language pair. Therefore, this paper aims to enhance the quality of the machine translation model by using the transformer model with a back-translation technique. An intermediate translation system was built using the bilingual dataset as a training corpus. This system was then used with a large monolingual dataset to generate the back-translation data, which can be considered as augmented training data for the translation model. The experimental result on the IWSLT'15 English-Vietnamese test set showed that the system with back-translation outperforms about 2.4 BLEU points than the system with the only transformer. With the test set of the Machine Translation shared task in VLSP 2020, the proposed system with back-translation was ranked as the first place with the highest score of human evaluation (1.55 points, compared to 1.33 points for second place). With the automatic evaluation, the system achieved a 32.1 BLEU score and a 0.50 TER score on VLSP 2020 Machine translation task test data.

1 Introduction

The demand for translation from one language to another is increasing due to the explosion of the Internet and the exchange of information between various regions using different regional languages. Machine translation has long been a major problem in the field of Natural Language Processing (NLP). Neural Machine Translation (NMT) has recently been put into research and has made huge improvements to machine translation systems. Most NMT systems are based on an encoder-decoder architecture consists of two neural networks (Bahdanau et al., 2016; Luong et al., 2015). The encoder compresses the source strings into a vector, used by the decoder to generate the target sequence. Sequence-to-sequence networks consist of two Recurrent Neural Networks (RNNs) and an attention mechanism has significant improvements compared to the traditional statistical machine translation approach.

To the best of our knowledge, transformer architecture networks have achieved the best results for many languages (Vaswani et al., 2017; Wang et al., 2019; Edunov et al., 2018). Transformer is a network architecture based on a self-attention mechanism. Transformers are good at machine translation and many NLP tasks because they totally avoid recursion, by processing sentences as a whole and by learning relationships between words thanks to multi-head attention mechanisms and positional embeddings. Recent networks include a number of parameters and they mostly focus on high-resource language pairs data.

However, those networks may not work well to low-resource translation tasks such as English-Vietnamese. Preparing a good quality bilingual data set is quite difficult, while the amount of monolingual data is quite abundant and available online. That raises a basic idea of using this single language data source to enhance the quality of the machine translation model. Some approaches to solving this problem include creating a language model to improve the quality of the machine translation model (Sennrich et al., 2016) or using back-translation.

In this paper, we propose a machine translation system participating in the Machine Translation Shared Task in VLSP 2020 (Thanh-Le et al., 2020). The main translation model in this system is the transformer with back-translation. This technique can be considered semi-supervised learning, whose

*Corresponding author

main purpose is data augmentation. Despite being simple, the back translation technique has achieved great improvements in both SMT (Bojar and Tamchyna) and NMT (Edunov et al., 2018).

The rest of this paper is organized as follows. Section 2 presents related works using encoder-decoder and back-translation architecture. Our methodology is presented in Section 3. The experiments are shown in Section 4 and Section 5. Finally, Section 6 concludes the paper and gives some perspectives for the work.

2 Related work

We build upon recent work on neural machine translation which is typically a neural network with an encoder/decoder architecture. The encoder represents information of the source sentence, while the decoder is a neural language model based on the output of the encoder. The parameters of both models are learned together to maximize the occurrence of target sentences with corresponding source sentences from a parallel corpus (Sutskever et al., 2014). At inference, a target sentence is generated by left-to-right decoding. Different neural architectures have been proposed with the goal of improving the efficiency of the translation system. This includes recurrent networks (Sutskever et al., 2014; Bahdanau et al., 2016; Luong et al., 2015), convolutional networks (Kalchbrenner et al., 2014; Gehring et al., 2017) and transformer networks (Vaswani et al., 2017). Recent work is based on the attention mechanism in which the encoder generates a sequence of vectors for each target token, the decoder pays attention to the most relevant part of the source through the weights of the vectors encoder (Bahdanau et al., 2016; Luong et al., 2015). Attention has been refined with self-attention and multi-head attention (Vaswani et al., 2017). The baseline model of our system is the transformer architecture (Vaswani et al., 2017).

The idea of back-translation has been suggested since statistical machine translation, where it was used for semi-supervised learning (Bojar and Tamchyna) or self-training (Vandeghinste, 2011). In the modern NMT study, (Sennrich et al., 2016) reported significant increases in terms of WMT and IWSLT shared tasks (Edunov et al., 2018), while (Currey et al., 2017) reported similar findings on low resource conditions, suggesting that even poor translations can make progress.

3 Methodology

3.1 Our proposed system architecture

Aforementioned, for the low-resource bilingual dataset like English-Vietnamese, we proposed to use the back-translation technique as an augmentation technique to build more data for the training corpus. Back-Translation can be considered as a semi-supervised learning technique. Firstly, an intermediate machine translation system is trained using existing parallel data. This system is used to translate the target to the source language. The result is a new parallel corpus in which the source side is a translation synthesizer while the target is the text is written by humans (monolingual dataset). Then, the synthesized parallel corpus is combined with the real text (bilingual dataset) to train the final system. Back-Translation does not need to change model architecture unlike using a language model. The basic idea to use the language model is scoring the candidate words proposed by the translation model at each time step or concatenating the hidden states of the language model and the decoder.

Figure 1 illustrates our proposed system architecture. In this paper, we adopted Transformer as the main translation model. Both monolingual and bilingual datasets must be cleaned and preprocessed before feeding to the Transformer model, which will be presented in subsection 3.2.

To build the final translation model, three main phases have to be performed:

- **Phase 1:** Training a Vietnamese-English translation model with transformer using the bilingual dataset.
- **Phase 2:** Generating an extra bilingual dataset from the monolingual dataset using the Vietnamese-English translation model in the previous phase. During this phase, we used greedy decoding to speed up the data generation process because the monolingual data set was quite large.
- **Phase 3:** Combining generated extra bilingual dataset with origin bilingual one and train the final Vietnamese-English translation model.

We use the same transformer architecture for the English-Vietnamese or Vietnamese-English translation model. Detail description of this architecture is presented in Subsection 3.3.

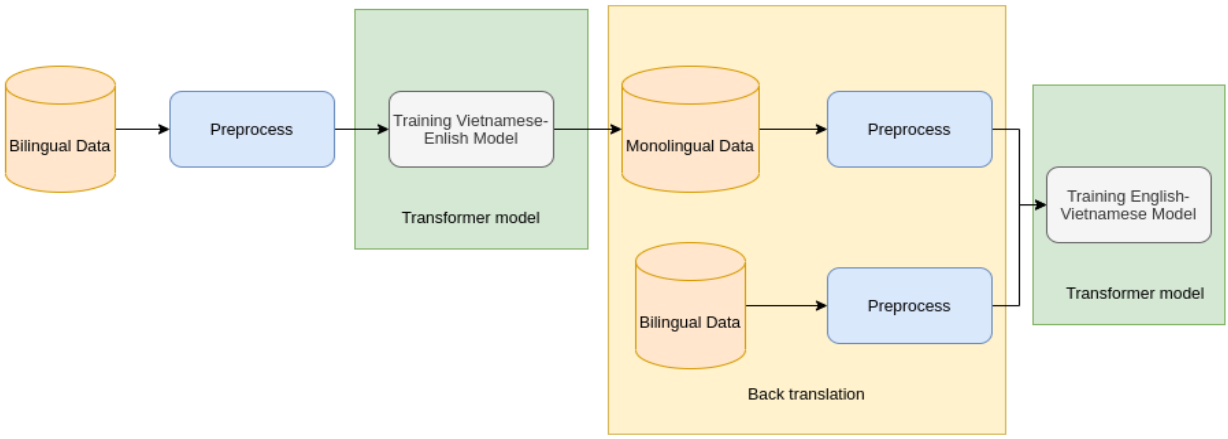

Figure 1: The proposed system architecture.

3.2 Text Pre-processing

3.2.1 VSLP 2020 Datasets

We received and only used two datasets from VLSP 2020 translation task (Thanh-Le et al., 2020) to develop our model. The monolingual dataset included about 20 million sentences crawled from a number of different e-newspapers. The bilingual database had about 4.14 million sentences from many different domains, presented in Table 1.

The bilingual dataset was used to train both English-Vietnamese and Vietnamese-English model while the monolingual dataset was used to create the back-translation dataset.

Table 1: The bilingual dataset on multi-domains

Dataset	Domain	Size (sentences)
News	News (in-domain)	20.0K
Basic	Basic conversations	8.8K
EVBCorpus	Mixed domains	45.0K
TED-like	EduTech talks	546.0K
Wiki-ALT	Wikipedia articles	20.0K
OpenSubtitle	Movie Subtitles	3.5M

3.2.2 Data cleaning and Pre-processing

The bilingual dataset was manually labeled by VLSP organizers so the problems with low translation quality are few. Therefore, we only need to remove too long sentence pairs in this dataset. All sentences having more than 250 words were eliminated.

Meanwhile, the monolingual dataset was crawled on the Internet. Therefore, this dataset had some problems in the raw text, e.g. too long sentences (due to the fault of the sentence tokenizer), non-Vietnamese language, HTML characters. We need a number of steps for data cleaning and preprocessing for this dataset. Some main steps were taken as follows.

- Removing non-Vietnamese sentences: Filtering out sentences that are not in Vietnamese using a language detection model;
- Removing sentences that are too long or too short;
- Cleaning HTML characters and some special characters.

After the data cleaning and pre-processing, the monolingual dataset had nearly 20 million remaining sentences, while the bilingual one had a total of 4.1 million sentence pairs. The data were cleaned, normalized, then lower-cased and tokenized using the Moses[1] tool. The data were learned a BPE set of 35,000 items using the Subword Neural Machine Translation toolkit[2].

3.3 The Transformer Model

The core idea behind the Transformer model is self-attention, the ability to attend to different positions of the input sequence to compute a representation of that sequence. The transformer creates stacks of self-attention layers to build both encoder and decoder instead of RNNs or CNNs. This general architecture helps transformer model calculated in parallel, instead of a series like RNNs, and learn long-range dependencies. The transformer architecture is presented in Figure 2.

Without the recurrence or the convolution, the transformer encodes the positional information of each input token by a position encoding function.

[1]Moses Open Source Toolkit for Machine Translation
[2]https://github.com/rsennrich/subword-nmt

Thus the input of the bottom layer for each network can be expressed as $Input = Embedding + PositionalEncoding$. The positional encoding is added on top of the actual embeddings of each word in a sentence.

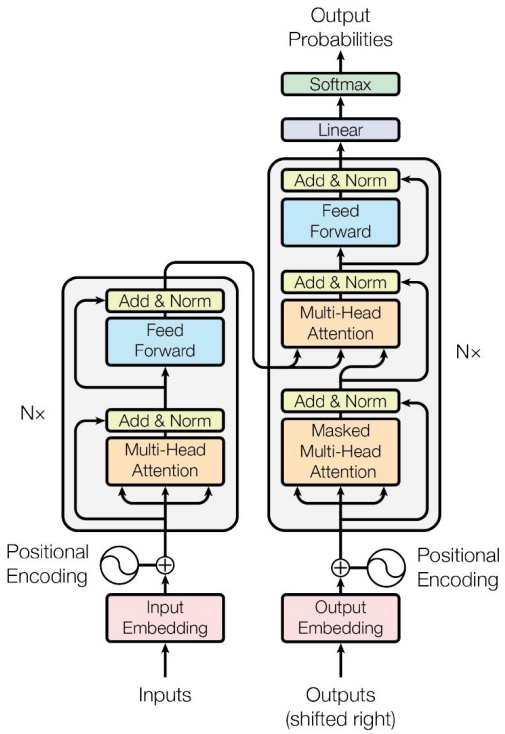

Figure 2: Transformer architecture.

The encoder has several layers stacked together. Each layer consists of a multi-head self-attention mechanism and a position-wise fully connected feed-forward network. Multi-head self-attention mechanism help model can pay "attention" to many certain pieces of content of the input.

The decoder is also a stack of identical layers, each layer comprising three sub-layers. At the bottom is a masked multi-head self-attention, which ensures that the predictions for position i depend only on the known outputs at the positions less than i. In the middle is another multi-head attention which performs the attention over the encoder output. The top of the stack is a position-wise fully connected feed-forward sub-layer. The decoder output finally goes through a linear transform with softmax activation to produce the output probabilities.

4 Experiment

4.1 Experimental setup

Transformer setup. We use the Transformer model in PyTorch from the fairseq toolkit[3]. All experiments were based on the Big Transformer architecture with 6 blocks in the encoder and decoder. We used the same hyper-parameters for all experiments, word representations of size 1024, feed-forward layers with inner dimension 4096. We used 16 attention heads, and we average the checkpoints of the last ten epochs. Models were optimized with Adam optimization using $\beta_1 = 0.9$, $\beta_2 = 0.98$, and $\varepsilon = 1e^{-}8$.

Back-translation set up. We run experiments on 2 GPU Tesla V100 and spent about 36 hours training the final model.

4.2 Automatic Evaluation and Human Evaluation

VLSP organizers provided two evaluation results for each model: (i) Automatic evaluation, and (ii) Human evaluation.

4.2.1 Automatic evaluation

In VLSP 2020, the automatic evaluation was used for reference, but not for the final decision for system ranking. The two metrics were BLEU and TER scores.

BLEU is a quality metric score for MT systems that attempts to measure the correspondence between a machine translation output and a human translation, as illustrated in Equation 1. The central idea behind BLEU is that the closer a machine translation is to a target human translation, the better it is.

$$\frac{\sum_{C \in Candidates} \sum_{ngram \in C} Count_{clip}(ngram)}{\sum_{C' \in Candidates} \sum_{ngram' \in C'} Count_{clip}(ngram')} \quad (1)$$

Translation Edit Rate (TER) is a method to determine the amount of Post-Editing required for machine translation jobs. The automatic metric measures the number of actions required to edit a translated segment inline with one of the reference translations, as illustrated in Equation 2.

$$TER = \frac{number\ of\ edits}{length\ of\ reference\ sentence} \quad (2)$$

[3]https://github.com/pytorch/fairseq

4.2.2 Human Evaluation

Human evaluation is the main metrics for ranking participating systems. There were 5 experts who were professional Vietnamese-English translators or interpreters. Each subject was asked to rate all systems from 1 to 6 based on Adequacy and Fluency. The overall rank was calculated by using the TrueSkill algorithm. TrueSkill is a rating system among game players. It was developed by Microsoft Research and has been used on Xbox LIVE for ranking and matchmaking services. This system quantifies players' TRUE skill points by the Bayesian inference algorithm.

5 Experimental Results

5.1 Experiment for Back-Translation

To find out the role of back-translation, we did some experiments on the IWSLT'15 English-Vietnamese test set. This test set is used from Stanford NLP group and has 1268 pairs Vietnamese-English sentence. Table 2 presents the results of the systems that used and did not use the back-translation (the baseline model with the transformer only). The experimental result showed that the model with back translation outperforms to the baseline one about 2.4% in BLUE score.

Table 2: Experimental results for the back translation on the IWSLT'15 English-Vietnamese test set

Model	BLEU score
Transformer (baseline)	36.3
Transformer + Back-Translation	38.7

5.2 VLSP 2020 Experimental Results

VLSP organizers released 2 test sets: a public test set and a private test set. The public test has 1220 pairs in the news domain while the private test is collected from online newspapers about Covid-19 news articles, about 789 pairs.

The result running on the private test is shown in Table 3. The final model that we submitted was our proposed system, which used Transformer and Back-Translation. Our system achieved a 32.10 BLEU score and a 0.5 TER score. According to the results of VLSP organizers, our BLEU score was at third and TER score is at second. However, the human evaluation of our system got the best result, which was 1.554. This led our system to be the first rank in the Machine Translation shared task in VLSP 2020.

As in Table 3, the automatic evaluation (BLEU) was on pair with human evaluation except in the case of our system. A possible reason was found that our system did not do casing recovery. The automatic evaluation metrics do consider casing, but the experts do not.

Table 3: Score of systems by VLSP organizer

Team	BLEU	TER	Human score
Our System*	***32.10***	***0.50***	***1.554***
EngineMT	38.39	0.45	1.327
RD-VAIS	33.89	0.53	0.864

5.3 Observations

After having some observations on the outputs of the baseline and the system with back translation, we find that the model using back translation gave more natural results than the baseline one.

For instance, as shown in the Table 4 by removing the duplicated pronounce "họ" (them) in the output, model using back translation avoids repeating words and makes the sentence more natural.

Table 4: Removing duplicated pronounces with back translation

Input: they will go back home to celebrate tet together with their families.
Baseline model: họ sẽ về nhà để ăn mừng tết với gia đình **họ**.
Baseline model + Back translation: họ sẽ trở về nhà để ăn mừng tết với gia đình.

With back translation, more suitable terms were selected in a specific context. As illustrated in Table 5, with the back translation mechanism, the "characteristics" word was translated into "đặc điểm" (properties), which suited best in the context. Whereas, the baseline model without back translation translated to "tính cách" (traits), typically one belonging to a person.

In addition, in some cases, back translation also helps the model generate some additional words, which can help to increase the fluency of the translation sentences (Table 6). This enhances the nat-

Table 5: More suitable terms with back translation

Input: typhoid's characteristics are continuous fever , high fever up to 40°C , excessive sweating , gastroenteritis and uncolored diarrhea.
Baseline model: **tính cách** của bệnh thương hàn là bệnh sốt liên tiếp, sốt cao lên đến 40 độ c, đổ mồ hôi quá nhiều, viêm dạ dày ruột và tiêu chảy không có màu.
Baseline model + Back translation: **đặc điểm** của bệnh thương hàn là sốt liên tiếp, sốt cao lên tới 40 độ c, đổ mồ hôi quá nhiều, viêm dạ dày và tiêu chảy không có màu.

Table 6: More natural expression with back translation

Input: thuan suggest to the delegation, in the short term to hurry up to prevent the epidemy, treat the disease, moreover, in the long term to make the whole team understand about malaria prevention method and therefore they will prevent disease for themselves which is also prevent disease for the whole team.
Baseline model: thuận gợi ý với phái đoàn, trong thời gian ngắn để nhanh chóng ngăn chặn sự phát bệnh, điều trị bệnh, hơn nữa, trong lâu dài để làm cho toàn bộ đội hiểu về phương pháp phòng ngừa bệnh sốt rét và do đó họ sẽ ngăn chặn bệnh này cho chính họ cũng sẽ ngăn chặn bệnh này cho cả đội.
Baseline model +Back translation: **ông** thuận gợi ý cho phái đoàn, trong thời gian ngắn để nhanh chóng ngăn chặn biểu mô, điều trị bệnh, hơn nữa, về lâu dài để cả nhóm hiểu về phương pháp phòng ngừa bệnh sốt rét và do đó họ sẽ ngăn ngừa bệnh tật cho bản thân, điều này cũng sẽ ngăn ngừa bệnh cho toàn đội.

uralness of the generated expression for the target language.

6 Conclusion

Participating in the machine translation shared task on VLSP 2020, we proposed some data cleaning and pre-processing for both monolingual and bilingual datasets. We did eliminate some very long or very short sentences as well as invalid characters (e.g. HTML, special ones). Some non-Vietnamese sentences in the monolingual dataset were also automatically removed. We proposed to use the transformer as the main translation model with back-translation as a data augmentation technique. An intermediate translation system was built using the bilingual dataset as a training corpus. The back-translation data were generated from the monolingual dataset by using the intermediate translation system. This back-translation data were then combined with the bilingual dataset to form the final training dataset for the final translation system.

The experiment results on the IWSLT'15 English-Vietnamese test set suggested that the back-translation is an effective data augmentation technique for deep learning machine translation models, which made an enhancement from 36.3 to 38.7 of the BLEU score. With the test set of Machine Translation shared task of VLSP 2020, this technique seemed can adapt quite well on the news domain. Our system with the back-translation technique was ranked as the first place with the highest score of human evaluation (i.e. 1.55 points, compared to 1.33 of the second place). With the automatic evaluation, the system achieved a 32.1 BLEU score and a 0.50 TER score on VLSP 2020 Machine translation task test data.

We will do some experiments on a number of sampling data methods during the preparation of back-translation datasets. We also consider analyzing and investigating the correspondences between human evaluation and automatic ones.

Ackknowledgement

This work was supported by the Vingroup Innovation Foundation (VINIF) under the project code DA116_14062019 / year 2019.

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2016. Neural Machine Translation by Jointly Learning to Align and Translate. *arXiv:1409.0473 [cs, stat]*. ArXiv: 1409.0473.

Ondr˘ej Bojar and Ales˘ Tamchyna. Improving Translation Model by Monolingual Data. page 7.

Anna Currey, Antonio Valerio Miceli Barone, and Kenneth Heafield. 2017. Copied Monolingual Data Improves Low-Resource Neural Machine Translation. In *Proceedings of the Second Conference on*

Machine Translation, pages 148–156, Copenhagen, Denmark. Association for Computational Linguistics.

Sergey Edunov, Myle Ott, Michael Auli, and David Grangier. 2018. Understanding Back-Translation at Scale. *arXiv:1808.09381 [cs]*. ArXiv: 1808.09381.

Jonas Gehring, Michael Auli, David Grangier, Denis Yarats, and Yann N. Dauphin. 2017. Convolutional Sequence to Sequence Learning. *arXiv:1705.03122 [cs]*. ArXiv: 1705.03122.

Nal Kalchbrenner, Edward Grefenstette, and Phil Blunsom. 2014. A Convolutional Neural Network for Modelling Sentences. *arXiv:1404.2188 [cs]*. ArXiv: 1404.2188.

Minh-Thang Luong, Hieu Pham, and Christopher D. Manning. 2015. Effective Approaches to Attention-based Neural Machine Translation. *arXiv:1508.04025 [cs]*. ArXiv: 1508.04025.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Improving Neural Machine Translation Models with Monolingual Data. *arXiv:1511.06709 [cs]*. ArXiv: 1511.06709.

Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. 2014. Sequence to Sequence Learning with Neural Networks. *arXiv:1409.3215 [cs]*. ArXiv: 1409.3215.

Ha Thanh-Le, Tran Van-Khanh, and Nguyen Kim-Anh. 2020. Goals, challenges and findings of the vlsp 2020 english-vietnamese news translation shared task. *Proceedings of the Seventh International Workshop on Vietnamese Language and Speech Processing (VLSP 2020)*.

V. Vandeghinste. 2011. Learning Machine Translation. * Cyril Goutte, Nicola Cancedda, Marc Dymetman, and George Foster. *Literary and Linguistic Computing*, 26(4):484–486.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention Is All You Need. *arXiv:1706.03762 [cs]*. ArXiv: 1706.03762.

Dilin Wang, Chengyue Gong, and Qiang Liu. 2019. Improving Neural Language Modeling via Adversarial Training. *arXiv:1906.03805 [cs, stat]*. ArXiv: 1906.03805.