A study of machine translation for Vietnamese and Korean on the TED Talks 2020 corpus

1st Binh Van Duong
University of Information Technology, VNU-HCM
18520505@gm.uit.edu.vn

4th Kim Chi T. Phan
University of Information Technology, VNU-HCM
18520525@gm.uit.edu.vn

3rd Chien Nhu Ha
University of Information Technology, VNU-HCM
18520527@gm.uit.edu.vn

5th Phat Cao Tran
University of Information Technology, VNU-HCM
18521233@gm.uit.edu.vn

6th Trong-Hop Do
University of Information Technology, VNU-HCM
hopdt@uit.edu.vn

***Abstract*—Vietnam has achieved impressive economic growth in the last two decades. It becomes a worth investing country in the area. Consequently, the need of understanding foreign investors from different countries (S. Korea in specific) is an essential issue. Therefore, building an automatic machine translation system with high precision is a necessary solution, especially during the COVID-19 pandemic, where keeping distance is the best way to avoid spreading the virus. As a result, this research presents some experimental results on the TED Talks 2020 dataset for the task Korean - Vietnamese and Vietnamese - Korean machine translation with the purpose of providing an overview of the dataset and a deep learning machine translation model for the problem.**

***Index Terms*—OPUS, Machine translation, Parallel corpus, Vietnamese-Korean.**

I. Introduction

Machine translation task is a conventional problem in deep learning research. There are many applications in real life, namely, education, tourism, intelligent virtual assistant, etc. Nevertheless, building a machine translation system to translate from Vietnamese to other languages is still a rough problem. Most of the previous research focuses on the Vietnamese - English, and English-Vietnamese translation task due to the rich resource of dataset [1, 2] and the popularity of English [3] and the difficulty in analyzing the language alike Korean language. Consequently, there is a small number of Vietnamese-Korean and Korean-Vietnamese machine translation research. Most of the research concentrate on building a parallel dataset without publishing the dataset. This leads to difficulty for other researchers wishing to test their methods on the dataset. It is challenging to develop a newly proposed method on the task due to the limit of the experimental dataset.

The research provides experimental results in order to assess the TED Talks 2020 [4] parallel dataset on the Vietnamese-Korean and Korean-Vietnamese machine translation task. This study gives the strong points of the dataset for developing an efficient automatic machine translation system. Korean is not based on the Latin alphabet, so it is difficult to read and write the language. Therefore, it needs an expert for the dataset analysis. The study presents experiments on different word segmentation based on the specific features of the language (i.e., Type 1, Type 2, Type 3, Type 4) with the state-of-the-art architecture (Transformer [5]). It hopes that this research can serve as a starting point for other researchers to develop their methods on the dataset.

The rest of this research is organized as follows: Section. II presents some previous studies of the problem. The dataset is described in Section. III. Methodologies and Experimental results are shown in Section. IV and V, respectively. Section. VI gives our conclusion and future directions for the following research.

II. Literature review

Nguyen et al. [6] released a publication in 2019. They contributed a large-scale parallel corpus for the Korean-Vietnamese machine translation task. The authors also applied the state-of-the-art Vietnamese word segmentation method RDRsegmenter to Vietnamese texts and UTagger to Korean texts during their process of research. In which, the word *eojeol*, a word unit delimited by space in a sentence, is morphologically analyzed before being fed into the machine translation system. This not only reduced the vocabulary but also increased the efficiency of the machine translation system. In comparison with the bi-directional long short-term memory (bi-LSTM) machine translation system and statistical machine translation system, the system fed by the input made from preprocessed data with Utagger and RDRsegementer gave the best results on the bi-LSTM machine translation system when translating from Korean to Vietnamese with BLEU = 27.79% and TER = 58.77%, from Vietnamese to Korean with BLEU = 25.44% and TER = 58.72%.

The performance of deep learning models applied to natural language processing such as recurrent neural networks (RNN), long short-term memory (LSTM), and gated recurrent unit (GRU) is not really efficient in terms of model training time

as well as experimental results for the machine translation problem. The main reason is the inconsistency of sentence length and some specific features of the linguistic typology described in [7]. Consequently, the introduction of transformer architecture that takes advantage of the position of each word and the model structure that receives all inputs at once brings benefits in not only training time but also the model's accuracy in the machine translation task. Although the transformer's performance is fundamentally better than other models, it takes a lot of time to adapt the model architecture to suit each specific task. Gao et al. [8] proposed a novel Scalable Transformers, which naturally contains sub-Transformers of different scales and have shared parameters. The performance obtained after testing with the En-Fr dataset with BLEU in the range of 42.8% to 43.3% is better than the traditional transformer architecture with BLEU 40.5%.

The exploration of data and computing power of computer systems brings a valuable resource for a lot of achievements in natural processing language tasks in general and machine translation in particular. Liu et al. [9] developed the mBART model and its versions mBART25 and mBART50 that are typical examples of taking advantage of multilingual corpora. The famous model architecture in recent years is the transformer, a multilingual pre-trained model, which could be referenced or fine-tuned for specific tasks.

III. Dataset

In this research, we utilized TED Talks 2020[4], a small part of one of the largest multilingual dataset (Open Parallel Corpus (OPUS)[10]) to implement several experiments. It contains a crawl of roughly 4,000 TED and TED-X transcripts from July 2020. All the transcripts were translated by a global community of volunteers from English to more than 100 languages. As TED Talks is famous for its vocabulary and context coverage as well as the purpose of this research, in this paper, we employed the Korean-Vietnamese version with the size of 323,525 parallel pairs.

In the dataset, there are many annotations for background noise in Vietnamese which is not relevant to the translated scripts. There are also some problems in translation quality in the dataset because it is voluntarily translated by communities. Subsequently, the quality of translated scripts needs to be noticed. As a result, we hired an expert to help assess the quality. This process is explained in detail in Section. V.

IV. Methodologies

A. Word segmentation methods

Although Korean is a synthetic language and Vietnamese is an analytic language, both have the same difficulty in determining word boundaries and word sense disambiguation.

Firstly, a sentence in Korean contains many *eojjeols*, a token unit delimited by whitespaces. However, an *eojjeol* also contains subtle parts, a word is a noun, a verb, or an adjective, and a following supplement to define the word's function in a sentence. Therefore, segmentation by *eojjeol* produces a high amount of vocabulary, but it needs a large corpus to cover all the cases of a word in the sentences.

Secondly, unlike English, a word in Vietnamese is not determined by whitespaces. Therefore, we segmented Vietnamese words using Pyvi with the Vietnamese tokenizer announced to have the f1_score = 0.985

Thirdly, most of the vocabularies of Korean and Vietnamese contain words stemming or borrowing from Chinese, particularly hanja for Sino-Korean and Hán_Việt for Sino-Vietnamese. Therefore, there are some similarity in the length and meaning of a word such as 가정 (ga-jeong) in Vietnamese contains two definitions gia_đình (family) and giả_định (assumption).

In this paper, we compared the differences between different word segmentation methods both in Korean and Vietnamese

- Type 1: Korean segmentation according to white spaces in a sentence (*eojjeol*) and Vietnamese segmentation by syllables.
- Type 2: Korean segmentation by *eojjeol* and Vietnamese segmentation by Pyvi[1].
- Type 3: Korean and Vietnamese segmentation by syllables.
- Type 4: Korean segmentation by syllables and Vietnamese segmentation by Pyvi.

B. Transformer architecture

Transformer is a a prominent architecture for constructing various state-of-the-art models: GPT-2, GPT-3, BERT, RoBERTa etc. we decide to choose Transformer stemmed from Pytorch to conduct our Korean - Vietnamese machine translation experiments. Here is some noticeable emphasises of a transformer network: Transformer architecture includes two main blocks as how numerous other sequence-to-sequence models are implemented which use up both input sentence and output sentence in the training process. Stacked encoders and stacked decoders are how transformer enhances its performance. *Nx* in Fig. 2 is representative for number of stacked encoders or encoders.

Input Embedding and Output Embedding.
After receiving encoded fixed-length pairs of sentences, the transformer carries out to embed these sentences into embedding matrices inside *Input Embedding* and *Output Embedding* as presented in Fig. 2. However, because transformer method is not trained based on time steps regarding other sequence to sequence models (RNN, GRU, LSTM). This leads to losing vital information of word or token positions in a sentence. Consequently, transformer calls for the support of *Positional Encoding* to fulfill the indispensable request of word orders.

[1]pyvi - https://pypi.org/project/pyvi/

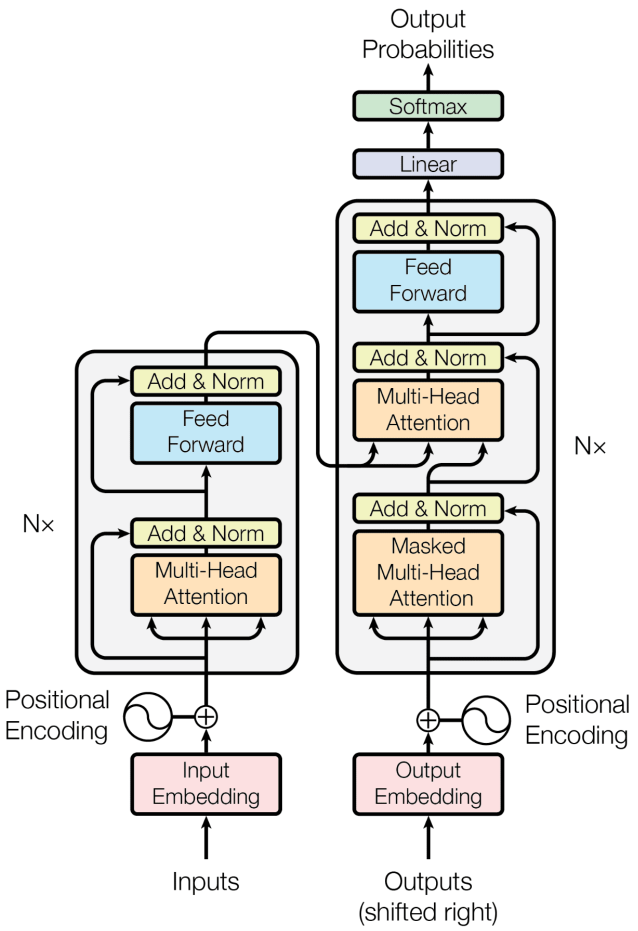

Fig. 1: Transformer architecture [11]

Positional Encoding.
Every word in a sentence will have a positional vector to indicate word order depending on the odd or even position it is located. Positional vectors have dimensions of 512, which is equal to the number of embedding to satisfy the following adding step.

$$PE_{(pos,2i)} = sin\left(\frac{pos}{10000^{2i/d_{model}}}\right), \quad (1)$$

$$PE_{(pos,2i+1)} = cos\left(\frac{pos}{10000^{2i/d_{model}}}\right), \quad (2)$$

Attention Mechanism.
Self-Attention: this is the heart of transformer architecture, boosting processing speed compared to traditional sequence to sequence models. This mechanism allows transformer to collect all encoded information simultaneously instead of using time steps. To model context for a sentence, each token will be considered in correlation with other tokens in the same sentence to form a word relationship in that sentence. Establishing a self-attention mechanism involves three elements: Query(Q), Key(K), and Value(V). The first step is to multiply each of the encoder input vectors with three weights matrices W(Q), W(K), and W(V) trained in the training process. This matrix multiplication will give us three vectors for each input vector: the key vector, the query vector, and the value vector. The second step in calculating self-attention is to multiply the Query vector of the current input with the key vectors from other inputs. Then, it is divided by the square root of the key vector dimension(d_k). Finally, the calculated result is taken into softmax before multiplying to the value vector. Here is the formula to better simulate how it works:

$$Attention(K,Q,V) = softmax\left(\frac{QK_i}{\sqrt{d_k}}\right)V. \quad (3)$$

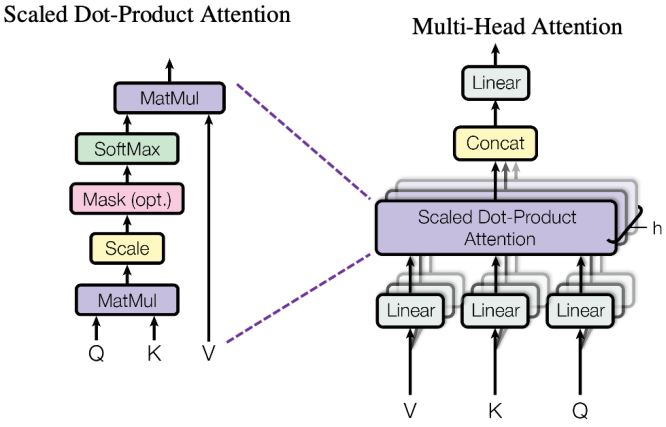

Fig. 2: Attention mechanism

Masked-Self Attention: This part is used in Decoder blocks and is similar to self-attention. However, it only permits target tokens at the present time-step to calculate attention weights using tokens from the previous time-step, without attention weights of after-positioned tokens.

Multi-head Attention: The essence of Multi-head Attention is stacking head attentions, in lieu of using only one self attention or one masked self attention. It helps transformer model utilize words relationship and better understand word orders in a sentence. After computing attention vector for each head, model will concatenate dot product of these heads and weight W.

$$head_i = Attention(Q_i, K_i, V_i) \quad (4)$$

$$Multi{-}head(Q,K,V) = Concat(head_1, ..., head_n)W \quad (5)$$

Feed Forward.
Synthesized information after adding and normalizing will come to Feed-Forward (dimension=2048), and its output and input will be combined. This step in the encoder returns context for the decoder. However, Feed-Forward in decoder

brings information to the Linear layer, and then softmax decides output sentences.

V. Experimental results

Data preprocessing. Due to the automatic process of crawling and preprocessing of Korean-Vietnamese TED Talks transcripts, manual corrections have been carried out. Although we have a large dataset, it still has special characters within the sentences, duplicate sentences in the meaning between two translations. Consequently, we carried out the manual correction in this dataset:

- Remove *nan* rows and duplicates
- Remove special characters within sentences
- Remove Korean sentences have *eojjeols* longer than 16 and corresponding Vietnamese sentences. The reason is the limitation of our computational system. In addition, when checking the lengths and meanings of long sentences, with the longest being approximately 150 *eojjeols*, we figured out the longer a sentence is, the worse the translation is. We hired one person who has TOPIK 4 certificate [12] for checking the liability of translations.

Due to the limitation of our bilingual checker, we randomly chose 100 sentences after deleting long sentences and checked the liability of those translations. Most of the examples were acceptable in the meaning side. In addition, as the translations are contributed by many volunteers, it has some different translation styles from Korean to Vietnamese and vice versa. We remained those sentences in this dataset. The final data size is 237,062 bilingual pairs. Then, we split the dataset into the ratio 8:1:1 corresponding to training, validation, test sets.

TABLE I: Vocabulary size of four different segmentation.

Word segmentation	Vocabulary size
Type 1	27,749
Type 2	1,754
Type 3	10,843
Type 4	6,376

Evaluation metrics.

- **BLEU score** [13] is a quality metric score for machine translation systems that attempts to measure the correspondence between a machine translation output and a human translation. The BLEU metric scores a translation on a scale of 0 to 100, with the purpose of measuring the sufficiency and naturalness of the machine translation output. The closer to 100 the test sentences score, the more similarity there is with their human target translations, and thus, the better the system is considered to be.
- **TER score** [14] stands for *translation edit rate* used for calculating the error rate of a machine translation output with a human translation based on the number of attempts needed to adjust the machine translation output to match exactly the meaning of the target sentence.

Results. When applying a model for translating from Vietnamese to Korean *eojjeol*, some model translations contain <UNK> label. This is because our corpus does not cover all functions of a word in a sentence going with different supplements, while this case is rarer in the model using Korean syllables segmentation, Type 3 and 4.

As Vietnamese does not have complicated grammar, hence, the translations from Korean to Vietnamese do not appear much <UNK> label. In addition, looking at translations have BLUE score higher than 30 and TER lower than 80, we recognized that although the model does not translate sentences as right as the reference translations, they have the same meaning. As examples are shown in Table. III, in fact, '보세요' and '볼까요' are exchangeable.

TABLE II: Results from different segmentation applied to the corpus using transformer model

Task	Word segmentation	Measurement (%)	
		BLEU	TER
Korean - Vietnamese	Type 1	30.51	84.99
	Type 2	**31.30**	86.66
	Type 3	29.64	95.81
	Type 4	31.23	**83.31**
Vietnamese - Korean	Type 1	24..50	94.38
	Type 2	23.84	94.28
	Type 3	27.11	82.47
	Type 4	**27.41**	**81.01**

VI. Conclusion and Discussion

In this study, we experimented on the TED Talks 2020 parallel pairs corpus with four different word segmentation methods. The results showed that this corpus still has many aspects that could be further explored. Especially, the style of translation and the situation of the scripts affect much to the performance of the model.

For future directions, the problem in the quality of translated scripts is not fully resolved. Therefore, the following research on the dataset could focus more on translation quality improvement and the different variations of transformer or any other methodologies that could be applied to obtain more comparative results. Besides, developing the methodologies on a specific situation of the translation scripts as well as different translation style could be gained more achievements.

References

[1] Long Doan et al. *PhoMT: A High-Quality and Large-Scale Benchmark Dataset for Vietnamese-English Machine Translation*. 2021. arXiv: 2110.12199 [cs.CL].

[2] Hong-Hai Phan-Vu et al. "Neural machine translation between Vietnamese and English: an empirical study". In: *Journal of Computer Science and Cybernetics* 35.2 (2019), pp. 147–166.

TABLE III: Translation examples

Task	Source	Target	Candidate
Korean - Vietnamese	다시 들어볼까요 (Let's listen again)	Hãy nghe lại nó lần nữa (Let's listen to it again)	Hãy quay lại lần nữa (Let's turn back again)
Vietnamese - Korean	Hãy nghe lại nó lần nữa (Let's listen to it again)	다시 들어볼까요 (Let's listen again)	다시 들어보세요 (Let's listen again)

[3] Lecturer/Doctoral Researcher Written by Kingsley Ugwuanyi. *A quarter of the world speak English - what makes it so popular?* URL: https://www.weforum.org/agenda/2020/02/english-dictionary-languages/#:~:text=a%20world%20language.-,English%20is%20now%20spoken%20by%20about%201.75%20billion%20people%20%E2%80%93%20a,all%20languages%20of%20the%20world..

[4] Nils Reimers and Iryna Gurevych. "Making Monolingual Sentence Embeddings Multilingual using Knowledge Distillation". In: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, Nov. 2020. URL: https://arxiv.org/abs/2004.09813.

[5] Ashish Vaswani et al. "Attention is all you need". In: *Advances in neural information processing systems*. 2017, pp. 5998–6008.

[6] Quang-Phuoc Nguyen et al. "Korean-Vietnamese Neural Machine Translation System With Korean Morphological Analysis and Word Sense Disambiguation". In: *IEEE Access* 7 (2019), pp. 32602–32616. DOI: 10.1109/ACCESS.2019.2902270.

[7] Shuoheng Yang, Yuxin Wang, and Xiaowen Chu. *A Survey of Deep Learning Techniques for Neural Machine Translation*. 2020. arXiv: 2002.07526 [cs.CL].

[8] Peng Gao et al. *Scalable Transformers for Neural Machine Translation*. 2021. arXiv: 2106.02242 [cs.CL].

[9] Yinhan Liu et al. *Multilingual Denoising Pre-training for Neural Machine Translation*. 2020. arXiv: 2001.08210 [cs.CL].

[10] *OPUS - an open parallel corpus*. URL: https://opus.nlpl.eu/index.php.

[11] Ashish Vaswani et al. *Attention is all you need*. https://arxiv.org/pdf/1706.03762.pdf. 31st Conference on Neural Information Processing Systems, Long Beach, CA, USA. 2017.

[12] *All About TOPIK Test - The Complete Guide: TOPIK GUIDE - The Complete Guide to TOPIK Test*. URL: https://www.topikguide.com/topik-overview/#:~:text=Test%20takers%20who%20meet%20the,high%20advanced%20(level%206)..

[13] Kishore Papineni et al. "Bleu: a method for automatic evaluation of machine translation". In: *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*. 2002, pp. 311–318.

[14] Matthew Snover et al. "A study of translation edit rate with targeted human annotation". In: *Proceedings of the 7th Conference of the Association for Machine Translation in the Americas: Technical Papers*. 2006, pp. 223–231.