Improving Quality of Vietnamese to Khmer Neural Machine Translation Using Multi-stage Fine-tuning Strategy

Tho Tran Duc[0009−0006−2773−9237], Huy Nguyen Trong[0009−0008−5598−9860] ⋆, and Huong Le Thanh[0000−0001−7298−1598] ⋆⋆

School of Information and Communication Technology,
Hanoi University of Science and Technology, Hanoi, Vietnam
{tho.td183634,huy.nt210451}@sis.hust.edu.vn, huonglt@soict.hust.edu.vn

Abstract. Machine translation for low-resource language pairs remains a challenging task, even with the advanced capabilities of large language models. Multilingual large language models require extensive monolingual data during the pre-training phase and large, high-quality parallel datasets for fine-tuning. In this paper, we present our research on an effective fine-tuning strategy for a pre-trained large language model to improve machine translation quality for the Vietnamese-Khmer language pair. Our experiments show that by applying self-supervised learning to the pre-trained model, fine-tuning it on related tasks, and then further fine-tuning it on both the original and augmented datasets, we achieved a BLEU score improvement of over 13% compared to the best results from previous studies and 7% higher compared to Google Translator and GPT-4 for in-domain tests.

Keywords: machine translation · low-resource · multi-stage fine-tuning · Vietnamese-Khmer

1 Introduction

In this increasingly interconnected world, language acts as both a bridge and a barrier. While it enables communication and understanding among diverse cultures, linguistic differences pose significant challenges, hindering effective interaction and cooperation. Machine Translation (MT), using computational algorithms, stands out as a powerful tool in overcoming these obstacles by performing the translation task among many languages. Thanks to the advancement of artificial neural networks, particularly the transformer architecture proposed by Vaswani et al. [1], Neural Machine Translation (NMT) has demonstrated superiority over previous machine translation methods in translation with higher quality. The transformer-based machine translation model utilizes self-attention

⋆ Corresponding Author
⋆⋆ Corresponding Author

mechanisms to weigh the significance of different words in a sentence, enabling it to capture long-range dependencies effectively.

However, low-resource language pairs translation remains a challenging task for researchers when building neural machine translation models. One of the main reasons is the limitation of available data for pre-training phase, as well as the scarcity of high-quality parallel dataset. Specifically, the case of Vietnamese-Khmer (Vi-Km) machine translation has showed little development. Both languages possess complex linguistic characteristics, which makes pre-trained neural machine translation models struggle to capture the representations without sufficient training data.

This paper aims to explore a comprehensive method to enhance the quality of Vi-Km neural machine translation. By leveraging the pre-trained SEA-LION developed by Ong and Limkonchotiwat [2], one of the first large language model for Southeast Asia countries, we propose several strategies to fine-tune this model with aiming to improve the quality of Vi-Km translation.

Our research has contributed a multi-step fine-tuning method that can be applied to improve the quality of machine translation for any low-resource language pairs. Our method leverages a moderately sized large language model, at the 3B parameter scale, which can be trained on free GPU environments such as Google Colab or in hardware-constrained settings. Additionally, the proposed method enhances machine translation quality by utilizing monolingual datasets and other tasks, without relying on large quantities of high-quality bilingual datasets.

2 Related Works

The challenge of NMT in low-resource settings has driven the development of several approaches aimed at overcoming the lack of large parallel corpora. One common approach is to leverage transfer learning, where a model pre-trained on high-resource languages is fine-tuned on low-resource language pairs. Zoph [12] demonstrated the effectiveness of transfer learning in low-resource NMT by pre-training a model on a high-resource language pair (e.g., English-German) before fine-tuning it on a low-resource pair (e.g., English-Turkish).

Another approach is data augmentation, where synthetic parallel data is generated to supplement the limited authentic data. Back-translation, introduced by Sennrich et al. [13] is one of the most successful techniques in this regard. In back-translation, target-side monolingual data is translated into the source language using a reverse translation model, and the generated synthetic parallel data is then used to train the forward translation model. This method has been widely adopted in low-resource NMT, including languages such as Vietnamese and Khmer ([14]).

A promising direction is the utilization of pre-trained multilingual models. These models, trained on massive amounts of data from multiple languages, have demonstrated impressive performance in various NLP tasks [16–18]. Fine-tuning a pre-trained multi-lingual model on a specific low-resource language pair is an advanced technique that has shown promising results. By leveraging the knowledge acquired from the pre-trained model, it is possible to improve the translation quality with limited training data. Liu et al. [17] demonstrated the effectiveness of leveraging large multilingual models like mBART, which was fine-tuned on specific language pairs to achieve significant improvements in translation quality. Pfeiffer et al. [18] also explored integrating domain-specific data during fine-tuning, further enhancing contextual accuracy for underrepresented languages in MT systems.

This paper builds on these advancements by proposing a novel method that integrates a multi-stage fine-tuning strategy with pre-trained multilingual models to enhance the quality of machine translation for low-resource language pairs. Our approach not only addresses the data scarcity issue but also seeks to optimize model performance in terms of both accuracy and efficiency.

3 Methodology

As mentioned above, the target of this paper is to improve the quality of the Vi-Km machine translation. Two most important factors to create a good machine translator are: (i) choosing a good pre-trained large language model (LLM); and (ii) determining a good fine-tuning strategy with suitable training datasets.

The SEA-LION [2] has been chosen as our base language model since it had been pre-trained on 980 billion text data from 11 languages used across Southeast Asia including Vietnamese and Khmer. The number of tokens used for training Vietnamese is 63.4 billion tokens (approximately 300GB of Vietnamese data), while the number of tokens used for training Khmer is 3.9 billion tokens (approximately 20GB of Khmer data). With the large amount of Vietnamese and Khmer data that the SEA-LION model has been pre-trained, the model has learned the features and representations of these two languages. In addition, the SEA-LION model was instruction-tuned on a wide range of tasks such as translation, summarization, and question answering, which helped the model better understand languages. By transferring the learned weights and features from the pre-trained model, we can take advantages of these parameters to speed up the fine-tuning process and enhance the Vi-Km machine translator's performance.

Currently, the SEA-LION model has two variants based on its size, including 3 billion and 7 billion parameters. Because of the limitation of hardware, the SEA-LION-3b with 3 billion parameters had been used in our experiments.

As far as we know, the largest bilingual Vi-Km dataset [7] includes of 137,175 sentence pairs in the training set, 2,000 sentence pairs in the validation set, and 2,000 sentence pairs for evaluation. This dataset will be used as the main dataset in our research.

To select an optimal fine-tuning strategy, we propose four distinct methods: (i) fine-tuning the base model using the original Vi-Km dataset [7]; (ii) fine-tuning the model with the augmented dataset; (iii) applying self-supervised learning to the SEA-LION model before fine-tuning it with the augmented dataset; and (iv) sequentially fine-tuning the model with a question-answering dataset followed by the augmented dataset. Detailed explanations for each proposed method are provided in the corresponding subsections of this paper.

3.1 Fine-tuning the Base Model with the Original Vi-Km Dataset

For baseline, we directly fine-tune the pre-trained SEA-LION-3b [2] model on original high-quality Vi-Km dataset. The SEA-LION model is a family of open-source language models developed by the AI Singapore research team with the aim of introducing a generative model for Southeast Asian languages. SEA-LION is built based on the MPT architecture [3] with 32 decoder layers and an input context length of up to 2048. By using the QLoRA [4] method, we can load this model onto low-hardware machine to commence its training stage.

3.2 Fine-tuning the Model with a Larger Dataset

To improving the model's generalization, we create an additional bilingual Vi-Km dataset by translating a monolingual Vietnamese dataset to Khmer by using back-translation method, following the method proposed by Senrich et al [5]. This dataset is mixed with the original Vi-Km dataset to fine-tune the SEA-LION model. The procedure to generate augmented data is outlined as follows:

- **Step 1 - Data collection**: We collect monolingual Vietnamese text data from the domain of storybooks[1], then proceed to segment them into sentences.
- **Step 2 - Bilingual data generation**: We use Google Translate API[2] to directly translate Vietnamese sentences collected from Step 1 to Khmer sentences.
- **Step 3 - Data filtering**: After generating synthetic bilingual dataset, we leverage Microsoft Translate API to translate the translated Khmer sentences back to Vietnamese. And then we compute TF-IDF [6] of the original and the back-translated Vietnamese sentences. After that, we compute cosine similarity score between two TF-IDF vectors of two sentences. Finally, the pairs of sentences with low similarity scores are filtered out.

Because of the complexity of lexical rules of both Vietnamese and Khmer, we utilise PyVi toolkit[3] to tokenize the Vietnamese sentences and Khmer-Nltk [8] to tokenize the Khmer sentences before transforming the sentences to TF-IDF vectors.

Data Filtering based on Similarity Score TF-IDF (Term Frequency-Inverse Document Frequency) helps determine the importance of each word in a specific document relative to the entire dataset. Words that appear frequently in

[1] https://www.kaggle.com/datasets/iambestfeeder/10000-vietnamese-books
[2] https://translate.google.com/
[3] https://pypi.org/project/pyvi/

one document but rarely in others are given higher weights, which is useful in information retrieval tasks to identify keywords or prominent content in the text. Besides, TF-IDF is a simple vectorization method, easy to compute, and time-efficient. It can be quickly applied to large datasets.

Assuming the document D consists of a list of sentences s, whereas each sentence s is a sequence of term t (tokens or words). Term frequency (TF) is the relative frequency of the appearance of term t in sentence s, which is formulated as in Equation 1.

$$tf(t,s) = \frac{f_{t,s}}{\sum_{t' \in s} f_{t',s}} \tag{1}$$

In Equation 1, $f_{t,s}$ is the frequency of term t in sentence s.

Inverse document frequency (IDF) measures the number of sentences in a set of document D that contain term t. The formula of IDF score is shown in Equation 2.

$$idf(t,D) = \log \frac{N}{1 + |s \in D : t \in s|} \tag{2}$$

In Equation 2, N is the number of sentences in document D; $|s \in D : t \in s|$ is the number of sentences in D containing t.

Under some cases, when the term t is not in D, which could lead to a division-by-zero, then this add 1 in the denominator of Equation 2.

The TF-IDF weight of a term t in the sentence s over the document D is calculated as in Equation 3.

$$w(t,s,D) = tf(t,s).idf(t,D) \tag{3}$$

The similarity score between two sentences is computed as the cosine similarity between two sentence's vectors. After calculating the similarity score for each pair of original Vietnamese sentence and the back-translated sentence, we decided to filter out the pairs which have cosine similarity scores lower than 90%. Retaining sentence pairs with a semantic similarity of over 90% ensures that the augmented sentence pairs have high quality and improve the performance of the model.

3.3 Self-supervised Learning and Fine-tuning the Model

Self-supervised learning is a continual pre-train method for large language models. Continual learning from specific-domain corpus on the previous pre-trained weights helps model retain the knowledge they learned and adapt more effectively on new domain. In this approach, we conduct a two-stage training of the base SEA-LION model as demonstrated in Figure 1. In the first stage, we perform self-supervised learning for the model on both the Vietnamese language and the target one to train the model understand these two languages more.

In the remaining stage, we fine-tune the self-supervised learning model using the mixed dataset of the original Vi-Km dataset and the additional bilingual Vi-Km one, as mentioned in the Section 3.2.

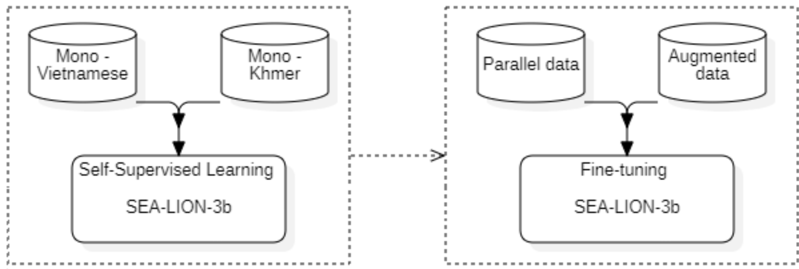

Fig. 1. Two-stage fine-tuning strategy

3.4 Sub-task Learning with E2E Question Answering

Since there are not much Vi-Km bilingual dataset, we investigate the impact of other datasets to the fine-tuning process of the machine translation task. In this research, the question answering dataset is used for this experiment. In the first step, we perform self-supervised learning as described in the previous Section 3.3 with more Vietnamese dataset driven from Zalo AI Challenge 2022[4], which includes monolingual Vietnamese sentences crawled from Wikipedia domain. Subsequently, we fine-tune the model on the E2E Question Answering subtask using the Zalo AI Challenge 2022 dataset[4]. Finally, we fine-tune the model on the machine translation task using the original Vi-Km dataset and the augmented dataset as described in the Section 3.2.

4 Experiments and Results

4.1 Datasets

VOV Dataset The main Vi-Km dataset is provided by Nguyen et al. [7], has been validated by language experts proficient in both Vietnamese and Khmer. The dataset has been pre-divided into four subsets with the following distribution: 137,175 sentence pairs for training, 2,000 sentence pairs for validation set, 2,000 sentence pairs within the news domain for evaluation (test set), and 2,000 sentence pairs outside the news domain for evaluation (out-of-domain test set). Since the dataset was collected from the VOV news station, it is referred to as the VOV dataset. The VOV dataset contains 137,175 samples in the training set, 2,000 in the validation set, 2,000 in the in-domain test set, and 2,000 in the out-of-domain test set.

From this dataset, we also extract the monolingual Vietnamese and Khmer sentences from the parallel data and use them in the self-supervised learning step mentioned in the Section 3.3. Consequently, we have a total of 137,175 monolingual Vietnamese sentences and 137,175 monolingual Khmer sentences.

[4] https://challenge.zalo.ai/

Synthetic Data Augmentation: With the method for data generation and filtering mentioned in Section 3.2, we collect totally 2,300 pairs of parallel Vi-Km sentences for data augmentation. This augmented dataset is then combined with the training data set of VOV dataset as in Section 4.1 for the fine-tuning phases in method in Section 3.2, Section 3.3 and Section 3.4.

E2E Question Answering - Zalo AI Challenge 2022 Dataset: This dataset is used to train the model for the method mentioned in Section 3.4. The data is extracted from the Zalo AI Challenge 2022 for the E2E Question Answering task. The dataset consists of two parts: a part including 237,422 monolingual Vietnamese sentences crawled from Wikipedia, which we use for continual self-supervised learning in Vietnamese, and the other remaining part comprises 3,000 samples for the E2E Question Answering task, with each sample containing a question, an answer, and an extracted passage.

4.2 Experimental Scenarios

For the experiment, we divided it into four test scenarios:

Scenario #1: This scenario implements sef-supervised fine-tuning original parallel dataset on the SEA-LION-3B model, which is available on Huggingface[5] using QLoRA mechanism [9] with an input sentence length of maximum 2048 tokens.

Scenario #2: This scenario, as proposed in Section 3.2, proceeds similarly to Scenario 1. After generating synthetic parallel data, we combine the augmented data with the original one to fine-tune the model.

Scenario #3: This scenario follows the proposed method in Section 3.3. In stage 1, the model undergoes self-supervised learning on the unlabeled Vietnamese and Khmer dataset. After that, the model is fine-tuned on the combination of the original and augmented dataset.

Scenario #4: This scenario, in detail, is described in Section 3.4. Firstly, the model undergoes self-supervised learning on the unlabeled dataset. In addition to the dataset used in Scenario #3, the model will also perform self-supervised learning on the Vietnamese Wikipedia dataset provided by Zalo as mentioned in Section 4.1. Then, we fine-tune the model on the question answering dataset. Finally, the model is fine-tuned in similar way as described in Scenario #2.

4.3 Experiment Setup

We conduct the experiment with SEA-LION-3B model using a free GPU Tesla P100, provided by Kaggle platform[6]. We used TRL, provided by Werra et al. [10] trainer API in Huggingface[7] for the training step. The detail statistics of hyperparameters using for training is as follow: $learning\_rate = 2e-4$,

[5] https://huggingface.co/aisingapore/sea-lion-3b
[6] https://www.kaggle.com/
[7] https://huggingface.co/docs/trl/index

$batch\_size = 2$, $input\_length = 2048$, $output\_length = 2048$, $weight\_decay = 0.001$, $warmup\_ratio = 0.03$ and $optim =$ '$paged\_adamw\_32bit$'.

Because the size of the base model can not be directly loaded and trained on the GPU memory. Thus, we apply QLoRA metdom, as proposed by Dettmers et al. [9] for parameter efficient fine-tuning on a P100. For quantization, we apply load-in 4-bit quantization, and the training regime is $float16$. And the LoRA configurations are: $lora\_alpha = 16$, $lora\_dropout = 0.1$, and $r = 64$.

Aiming at designing appropriate prompts that instruct the model follow strictly, we use instruction prompting in which tasks are described specifically and directly within the prompt itself. The template of the prompts is described in the Table 1.

Table 1. Prompting template

#Instruction:
Translate from Vietnamese to Khmer
#Input: <Input sentence>
#Output: <Output sentence>

To evaluate the effectiveness of our experiments, we use the BLEU score, proposed by Papineni et al. [11] for each pair the target and translated Khmer sentences. Before calculating the BLEU score, the translated sentences are tokenized by using khmer-nltk, by Hoang [8].

4.4 Experimental Results

We evaluated our different proposed scenarios on a test set with 2,000 parallel sentence pairs from in-domain test set and 2,000 samples from out-domain test set driven from VOV dataset in Section 4.1. Table 2 shows the results of our each training scenario on both in-domain and out-domain tests.

Table 2. BLEU scores on test set and out-domain test set of each training scenario

Scenario	**In-domain Score (%)**	**Out-domain Score (%)**
#1 Fine-tuning on the base model	51.11	44.87
#2 Fine-tuning with data augmentation	52.31	45.92
#3 Self-supervised learning with fine-tuning	56.21	46.37
#4 Sub-task learning with E2E QA	**56.99**	**47.00**

The results from the Table 2 show that by fine-tuning directly on the SEA-LION-3B model with the original bilingual dataset, we achieved a BLEU score of 51.11% on the in-domain test set and 44.87% on the out-of-domain test set. Moreover, by augmenting data, the score on in-domain test set and out-domain test set increased by 1.20% and 1.05% respectively, concluding that adding more parallel synthetic data can help the model generalize better. Noticeably, self-supervised learning before fine-tuning showed significant improvements on the in-domain test set with a BLEU score of 56.21%, which is 5.10% compared to the baseline and on the out-of-domain test set with 46.37%. Besides, learning an additional task, which is Question Answering, before fine tune on translation task, the model achieved the highest score with 56.99% and 47.00% BLEU score on in-domain and out-domain test set consequently.

These results indicate that continual pre-training with monolingual data considerably boosts the natural language inference and generation capabilities of large language models, thereby improving performances in multilingual tasks, including machine translation. This approach is even more effective when applied to low-resource languages, as these languages typically have far fewer pre-training resources compared to more widely spoken languages. In addition, training with the E2E Question Answering task also slightly improved the BLEU score. However, during experiments, due to the relatively small number of samples collected and used for training, the effectiveness of the question answering task has not been clearly demonstrated.

In addition to the experiment results, our research also recorded the training time (measured in hours) for each method. Specifically, Scenario #1 required 47 hours, Scenario #2 took 51 hours, Scenario #3 required 160 hours, and Scenario #4 took the longest at 288 hours. From Table 2 and the training time information, we can also conclude that there is a trade off between BLEU score and training time. By adding more stage in fine-tuning phase can help model to improve the performance, while the cost for training time is more expensive. During the evaluation process, we observed that, the inference time for each sentence with an average of 50 tokens takes from 5 to 6 seconds on a P100 GPU. With such that inference time, it is possible to serve the model in reality that meets the accuracy and speed requirements of users.

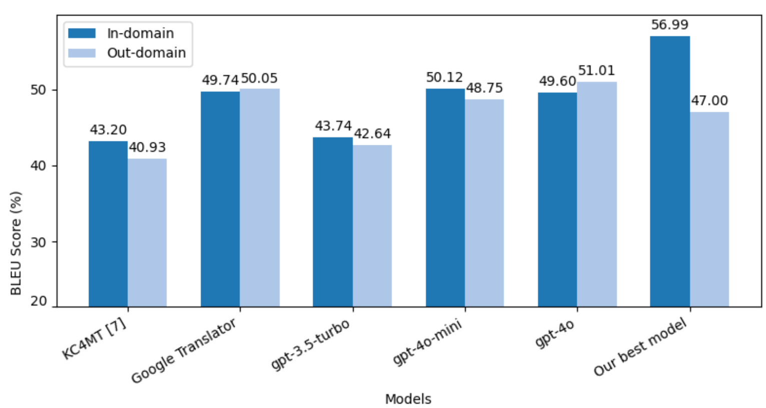

Fig. 2. Comparisons of our works to the previous research, other Large Language Models and translation tools

In addition to the results of the above training scenarios, we also conducted in-domain and out-of-domain tests on the prior work, KC4MT [7] by Nguyen et al., as well as on other large language models and translation tools, including GPT-4o, GPT-4o-mini, GPT-3.5-turbo[8], using OpenAI key, and Google Translator. For GPT-family models, we generated translated sentences by using proper zero-shot prompting. Then, we compared our best model to these models by using BLEU score, which is shown in Figure 2.

From the results in Figure 2, our highest scores improved by 13.79% in-domain test result, and 16.07% out-domain test result compared to previous work from KC4MT research by Nguyen et al. [7]. It is also shown that our model achieved the highest score for in-domain test of translating from Vietnamese to Khmer sentences. For out-of-domain test, gpt-4o had the highest BLEU score of 51.01%, approximately 4% higher than our best model. When comparing on the in-domain test set, it can be observed that the gpt-4o-mini model performed better with straightforward tasks in specific domains. Meanwhile, the gpt-4o model, with its better scalability, performed better in larger domains.

Our model outperforms the GPT models in in-domain tests but falls short in out-domain tests because it was trained on a dataset specific to the in-domain test set. Consequently, its performance declines when applied to out-domain data. In contrast, GPT models, having been trained on a larger and more diverse dataset, are better equipped to handle out-domain scenarios.

5 Conclusions

This study explored the challenges of machine translation for low-resource language pairs, focusing on the case of Vietnamese-Khmer. We discussed the limitations and obstacles identified in previous research and the available datasets aimed at addressing this issue. Consequently, we proposed multi-stage training methods for machine translation using the large language model SEA-LION. Our results demonstrate that the strategy of self-supervised learning on monolingual Vietnamese and Khmer corpus, followed by self-supervised fine-tuning joint-task including question answering and translation with data augmentation, yielded the best outcomes. We achieved a BLEU score of 56.99% on the in-domain test set and 47.00% on the out-of-domain test set, marking significant improvements of 13.79% and 16.07%, respectively, over the best previous research. Additionally, our approach outperformed Google Translator and gpt-4o by approximately 7% on an in-domain test set. These findings suggest that translation quality in large language models can be enhanced through a proper multi-stage fine-tuning pipeline that combines continual pre-training, joint-task learning, and data augmentation. For future work, we plan to integrate additional tasks or explore other continual pre-training methods within our training stages to further enhance model performance.

[8] https://platform.openai.com/docs

References

1. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A., Kaiser, Ł. & Polosukhin, I. Attention is All you Need. *Advances In Neural Information Processing Systems*. **30** (2017)
2. Ong, D. & Limkonchotiwat, P. SEA-LION (Southeast Asian Languages In One Network): A Family of Southeast Asian Language Models. *Proceedings Of The 3rd Workshop For Natural Language Processing Open Source Software (NLP-OSS 2023)*. pp. 245-245 (2023,12), https://aclanthology.org/2023.nlposs-1.26
3. Team, M. Introducing MPT-7B: A New Standard for Open-Source, Commercially Usable LLMs. (2023), www.mosaicml.com/blog/mpt-7b, Accessed: 2023-05-05
4. Dettmers, T., Pagnoni, A., Holtzman, A. & Zettlemoyer, L. QLoRA: Efficient Finetuning of Quantized LLMs. *ArXiv Preprint ArXiv:2305.14314*. (2023)
5. Sennrich, R., Haddow, B. & Birch, A. Improving Neural Machine Translation Models with Monolingual Data. *In Proc. of ACL 2016*. pp. 86-96.
6. Baena-Garcia, M., Carmona-Cejudo, J., Castillo, G. & Morales-Bueno, R. TF-SIDF: Term frequency, sketched inverse document frequency. *11th International Conference On Intelligent Systems Design And Applications*. pp. 1044-1049 (2011)
7. Nguyen, V., Nguyen, H., Le, H., Nguyen, T., Bui, T., Pham, L., Phan, A., Nguyen, C., Tran, V. & Tran, A. KC4MT: A High-Quality Corpus for Multilingual Machine Translation. *In Proc. of LREC 2022*. pp. 5494-5502.
8. Hoang, P. Khmer Natural Language Processing Tookit. *GitHub Repository*. (2020), https://github.com/VietHoang1512/khmer-nltk
9. Dettmers, T., Pagnoni, A., Holtzman, A. & Zettlemoyer, L. Qlora: Efficient finetuning of quantized llms. *ArXiv:2305.14314*. (2023)
10. Werra, L., Belkada, Y., Tunstall, L., Beeching, E., Thrush, T., Lambert, N. & Huang, S. TRL: Transformer Reinforcement Learning.(2020), https://github.com/huggingface/trl
11. Papineni, K., Roukos, S., Ward, T. & Zhu, W. Bleu: a method for automatic evaluation of machine translation. *In Proc. of ACL'02*. pp. 311-318 (2002)
12. Zoph, B., Yuret, D., May, J. and Knight, K. Transfer learning for low-resource neural machine translation. *In Proc. of EMNLP 2016*. pp. 1568-1575.
13. Sennrich, R., Haddow, B., and Birch, A. Improving neural machine translation models with monolingual data. *In Proc. of ACL 2016*. pp. 86-96.
14. Nguyen Quoc, T., Le Thanh, H. & Pham Van, H.. Low-Resource Neural Machine Translation Improvement Using Data Augmentation Strategies. *Informatica (Slovenia)*. 47(3) (2023)
15. Lample, G., Conneau, A. Denoyer, L., Ranzato, M. Unsupervised machine translation using monolingual corpora only. *ArXiv Preprint ArXiv:1711.00043*. (2018)
16. Kozhirbayev, Z. Enhancing neural machine translation with fine-tuned mBART50 Pre-Trained model: An examination with low-resource translation pairs.. *Ingénierie Des Systèmes D'Information*. **30** pp. 831-838 (2024)
17. Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L. & Stoyanov, V. Multilingual Denoising Pre-training for Neural Machine Translation. *In Proc. of ACL 2020*. pp. 7260-7274.
18. Pfeiffer, J., Rücklé, A., Poth, C., Kamath, A., Rothermel, M. & Gurevych, I. AdapterFusion: Non-Destructive Task Composition for Transfer Learning. *Findings Of The Association For Computational Linguistics: EMNLP 2022*. pp. 270-282 (2022)