Towards Improving Low-Resource Machine Translation with Lightweight Training and Synthetic Data: Case Study of Vietnamese - Khmer

Trong Huy Nguyen, Thanh Huong Le, Que Nhu Tran

Proceedings of the 39th Pacific Asia Conference on
Language, Information and Computation (PACLIC 39)

Emmanuele Chersoni, Jong-Bok Kim (eds.)

2025

Towards Improving Low-Resource Machine Translation with Lightweight Training and Synthetic Data: Case Study of Vietnamese - Khmer

Trong Huy Nguyen[1], Thanh Huong Le[1] *, Que Nhu Tran[2],
[1]Hanoi University of Science and Technology, Vietnam
[2]Foreign Trade University, Vietnam
Huy.NT210451@sis.hust.edu.vn, huonglt@soict.hust.edu.vn,
trannhuqueht@gmail.com

Abstract

Neural machine translation for low-resource languages remains challenging due to the scarcity of high-quality parallel corpora and community support. Under the view of Vietnamese-Khmer (Vi-Km) translation task, we focus on developing a comprehensive, lightweight training pipeline that minimizes reliance on extensive parallel data and large-scale model architecture, while achieving substantial performance improvements. Then, we introduce a three-stage training strategy built upon a small pre-trained language model: (1) Small-scale Continual Pre-training on monolingual data, (2) Supervised Fine-Tuning on parallel datasets and synthetic data generated through a novel data augmentation framework, and (3) Direct Preference Optimization leveraging a newly constructed preference-ranking dataset guided by LLMs. Experimental results on the VOV dataset demonstrate that our model significantly outperforms by up to 8% in both directions on BLEU & METEOR scores over strong LLMs, and other commercial systems. These findings confirm the effectiveness of our approach for enhancing low-resource machine translation systems under low-cost computation resources and the reproducibility for other underrepresented languages.

1 Introduction

In the modern era, with thousands of languages worldwide, machine translation (MT) has become a core focus of Natural Language Processing (NLP) research. With the recent advent of pre-trained language models (PLMs) with the self-attention mechanism (Vaswani et al., 2017), machine translation quality has seen significant improvement, even in low-resource languages. At the same time, large language models (LLMs) have simplified the modeling of complex grammatical and semantic relationships between languages, resulting in more fluent and coherent translations (Zhu et al., 2024). However, both PLMs and LLMs rely on massive, high-quality corpora and computational budgets that are rarely available for extremely low-resource languages. Current research aims to address the challenges by developing a training methodology that minimizes dependence on extensive data and human resources while retaining the strengths of precedent approaches. Meanwhile, some of the latest research is shifting towards data augmentation by leveraging available monolingual data (Sennrich et al., 2016; Liu et al., 2021) or generating synthetic bilingual data. Another common method is translating through a pivot language (Kim et al., 2019). However, these approaches still face limitations in data quality, scalability, and computational requirements (Edunov et al., 2020).

Given the limited resources available for the case of Vi-Km translation and the impracticality of deploying large models in resource-constrained environments, there is a compelling need for an efficient, lightweight, and scalable training pipeline tailored to this specific task. Furthermore, existing few-shot prompting methods on LLMs struggle to deliver competitive results for this pair, emphasizing the importance of task-specific model adaptation and domain-specific fine-tuning.

The problem we address is how to leverage small language models with available datasets using an effective training approach while ensuring compatibility with limited computation infrastructure and involvement of human annotators. We propose a novel comprehensive approach for low-resource MT, with the following key contributions:

- A lightweight training pipeline, consisting of Small-scale Continual Pre-training (SmCPT), Supervised Fine-Tuning (SFT), and Direct Preference Optimization (DPO) (Rafailov et al., 2023), optimized for small language models on limited computational resources.

* Corresponding author: huonglt@soict.hust.edu.vn

- A scalable synthetic parallel data generation framework leveraging powerful LLMs with filtering and ranking mechanisms to construct high-quality training data, including preference-ranking datasets for DPO, without manual annotation.

- An extensive empirical study on the VOV dataset (Nguyen et al., 2022), achieving the highest results on BLEU and METEOR scores over previous works, commercial translation systems, and LLMs.

This study provides practical insights and scalable solutions for improving low-resource MT and paves the way for extending these techniques to other under-supported language pairs in the region.

2 Related Works

In the scope of addressing low-resource MT, one notable approach is pre-training a multilingual model using Transformer Encoder-Decoder (Xue et al., 2021). The model is then fine-tuned for many specific languages, showing prominent performances as in the cases of Kazakh-Russian and Russian-Tatar (Kozhirbayev, 2024). Data augmentation is also considered an effective approach. Sennrich et al. (2016) proposed back-translation, involving translating target monolingual data into the source language using a reverse translation model. The synthetic parallel data generated through back-translation is subsequently employed to train the forward translation model. This method has shown improvements in low-resource NMT, notably in the Vi-Km (Pham Van and Le Thanh, 2022; Quoc et al., 2023) research. Besides, employing paraphrase embedding and POS-Tagging is an efficient approach to augment data for machine translation by paraphrasing sentences at the word level (Maimaiti et al., 2021). However, these approaches have revealed several limitations, especially the loss of contextual information (Edunov et al., 2020). Using pivot languages is another solution to the challenges of machine translation proposed by Kim et al. (2019). The selected pivot language data is used to train a cross-lingual encoder and auto-decoder, which is then fine-tuned to produce the final translation model. Kim et al. (2019) has demonstrated the effectiveness of this method in the French-German and German-Czech translation tasks in the WMT 2019. However, this approach for low-resource machine translation has several disadvantages; notably, it often leads to the propagation of circular translation errors (Kementchedjhieva and Søgaard, 2023).

KC4MT (Nguyen et al., 2022) provides a high-quality Vi-Km bilingual dataset curated from Voice of Vietnam (VOV) news content. It was developed under a national project and validated by language experts proficient in both Vietnamese and Khmer. Besides the Vi-Km bilingual dataset, this project released an additional monolingual Khmer subset. In addition to KC4MT, the TED (Reimers and Gurevych, 2020) dataset offers a monolingual source for Vietnamese and Khmer. TED includes transcripts from over 4,000 TED and TEDx talks translated by a global volunteer community into more than 100 languages. QED (Abdelali et al., 2014), developed by the Qatar Computing Research Institute, provides multilingual subtitles from educational content across various STEM topics. While these corpora offer valuable resources, previous studies have noted challenges related to the presence of non-linguistic noise, such as special characters or inconsistent formatting.

3 Methodology

3.1 Backbone Model Selection

In this study, we leverage a pre-trained language model that offers strong multilingual support for both Vietnamese and Khmer. Among the available models, SeaLLMs (Nguyen et al., 2024) stands out as a family of large language models specifically designed for Southeast Asian (ASEAN) languages. These models are fine-tuned and optimized to enhance accessibility and performance for low-resource regional languages across ASEAN. The third version, `SeaLLMs-v3`, offers two new variants of size: 7B and 1.5B. Compared to its predecessors, `SeaLLMs-v3` is pre-trained on a massive corpus comprising general-domain and region-specific data, including sources such as Wikipedia, CC-News, CulturaX, and MADLAD-400 (Nguyen et al., 2024; Kudugunta et al., 2023). It also incorporates synthetic and translated data in training to improve support for under-supported languages in the region.

In this work, we select `SeaLLMs-v3-1.5B-Chat` (Nguyen et al., 2024) as our baseline model, a further instruction-tuned version of the 1.5B base model. This choice is motivated by its balance between model architecture and computational efficiency due to its small size, offering an effective foundation for our experiments.

3.2 Training Methodology

To address the challenges of MT for the low-resource Vi–Km language pair, we propose a multi-stage training framework tailored for a lightweight model architecture. This approach integrates several optimization techniques to balance performance and computational efficiency. The overall framework is illustrated in Figure 1, consisting of three sequential stages:

1. **Small-scale Continual Pre-training phase**: The baseline model is continually pre-training on monolingual data using an adapter-based architecture, enabling better language modeling capacity specific to the target languages.
2. **Supervised Fine-Tuning phase**: The continual pre-trained model is then fine-tuned on a mixture of real and synthetic parallel data to enhance translation quality under limited supervision.
3. **Direct Preference Optimization Training phase**: The model generates multiple candidate translations from the source side of a gold-standard parallel dataset. These candidates are then ranked by a large language model (LLM), and the resulting preference pairs are used to train the model with a direct preference optimization objective.

For adapter type selection, we revisit two low-rank adapter variants—LoRA (Hu et al., 2022) and AdaLoRA (Zhang et al., 2023) by fine-tuning on the original VOV Vi-Km dataset. The adapter configuration that yields the highest validation performance is carried forward to all subsequent stages. After completing the full training pipeline for the Vi-Km translation direction, we reapply the same experimental stages to the reverse direction (Km-Vi). The adapter configuration, hyperparameters, and datasets are maintained to ensure a fair comparison. This bidirectional evaluation enables a more comprehensive assessment of our pipeline's scalability and its effectiveness applied to another low-resource pair.

For automatic evaluation, we use the BLEU (Papineni et al., 2002) and METEOR (Lavie and Agarwal, 2007) metrics, which are widely used in machine translation tasks. As both metrics rely on n-gram overlap, we employ the `khmer-nltk`[1] and `pyvi`[2] for Km and Vi tokenizers, respectively.

3.2.1 Small-scale Continual Pre-training

This stage adapts the baseline PLM to low-resource languages by conducting continual pre-training on monolingual non-labeled data. To ensure computational efficiency, it updates only a small subset of parameters — specifically, the adapter modules, embedding layers, and task-specific heading layer — while keeping the rest of the base model frozen.

This approach addresses the issue of unstable training typically caused by random initialization of adapter parameters, leading to a negative effect on model performance (Nguyen and Nguyen, 2025). The objective, as shown in Equation 1, only updates the adapter parameters ϕ and freezes model parameters θ, while optimizing the negative log-likelihood of the next token given the preceding context.

$$\mathcal{L}_{\text{NLL}}(\mathbf{x}, \phi) = -\sum_{t=1}^{T} \log P(\mathbf{x}_t | \mathbf{x}_{<t}; \theta, \phi) \quad (1)$$

The data for SmCPT are extracted from both bilingual and monolingual sources. Let $\mathcal{D}_{\text{bi}} = \{(x_i, y_i)\}_{i=1}^{N}$ denote a bilingual dataset of N pairs in train set, where $x_i \in \mathcal{L}_s$ (source language) and $y_i \in \mathcal{L}_t$ (target language). From this, two monolingual subsets are extracted: $\mathcal{D}_s = \{x_i\}$ and $\mathcal{D}_t = \{y_i\}$. Additionally, external monolingual corpora are collected independently for each language, denoted as $\mathcal{D}'_s \subset \mathcal{L}_s$ and $\mathcal{D}'_t \subset \mathcal{L}_t$. The final dataset used for SmCPT is constructed by merging all these sources: $\mathcal{D} = \mathcal{D}_s \cup \mathcal{D}_t \cup \mathcal{D}'_s \cup \mathcal{D}'_t$. This composition enables the model to benefit from both translation-aligned data and naturally occurring monolingual text during continual pre-training.

3.2.2 Supervised Fine-Tuning

After SmCPT, the model is fine-tuned using the parallel dataset to further adapt to the translation task. Besides, we also propose a synthetic data generation pipeline utilizing power from a strong LLM: `GPT-4o` to enrich the parallel dataset. The synthetic dataset is mixed with the training set of the VOV dataset to supervise fine-tuning the model. Here, we use a fixed prompt during the training translation task, and the representations of the prompt were not updated in the loss function (Zhang et al., 2024). The loss function is represented in Equation

[1] https://github.com/VietHoang1512/khmer-nltk

[2] https://github.com/trungtv/pyvi

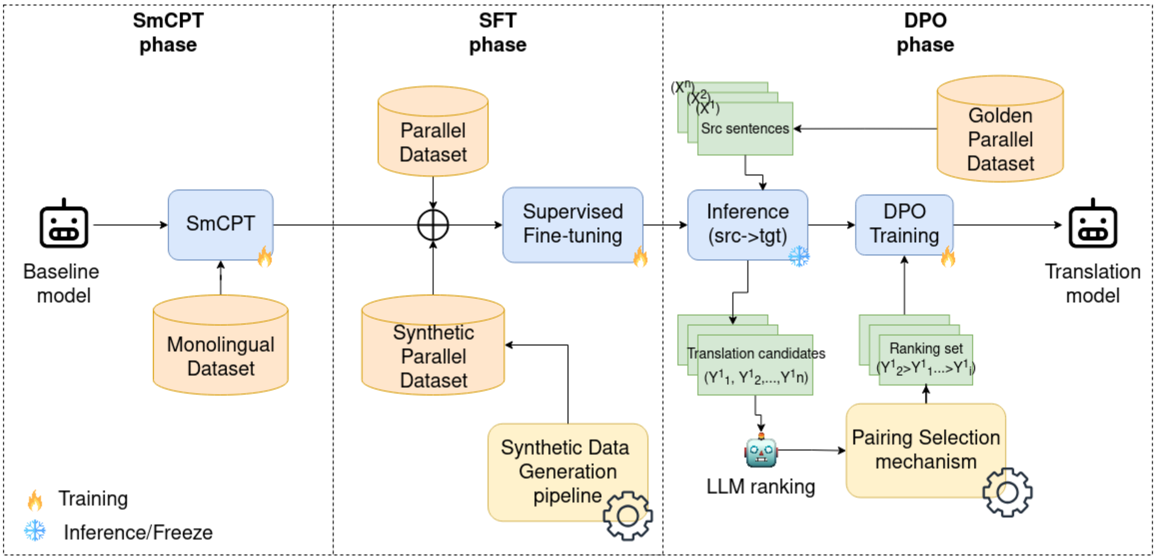

Figure 1: Our proposed training pipeline

2, in which the model parameters ϕ are updated based on the negative log-likelihood loss of the next token prediction and frozen the fixed prompt $\mathcal{I}$ and the base θ.

$$\begin{aligned}\mathcal{L}_{\text{NLL}}(\mathbf{x}, \mathbf{y}, \phi) &= -\log P(\mathbf{y}|\mathbf{x}; \mathcal{I}, \theta, \phi) \\ &= -\sum_{t=1}^{T} \log P(y_t|y_{<t}, \mathbf{x}; \mathcal{I}, \theta, \phi)\end{aligned} \tag{2}$$

3.2.3 Direct Preference Optimization

The fine-tuned model generates candidate Khmer translations from a golden subset of 2,000 Vietnamese sentences in the bilingual datasets of the VOV. These candidates are then ranked by `GPT-4o` with the provided target Khmer sentence serving as 'golden translation' to create a preference dataset in the form of pairing samples, which is inspired by the idea of *self-knowledge* (Yang et al., 2024), the student learns from its outputs and preference ranking, evaluated by a teacher model. Experiments by Yuan et al. (2024) show that models are capable of self-alignment via LLM-based judging and training on their generations through iterative DPO training. However, when forming sentence pairs for DPO training, some pairs of responses with nearly the same representations could end up being separated far apart during DPO training, leading to potential bias in the model (Yan et al., 2025). Thus, we also propose a selection mechanism to form a high-quality DPO pair dataset and control the strictness of the loss function based on β hyperparameter, as illustrated in Equation 3 (Rafailov et al., 2023), which is later shown in Appendix B.3. In which, β is a temperature parameter that controls the sensitivity to differences in rewards, $\pi_\theta(y|x)$, which is the probability of the model generating response y given input x and $\pi_{ref}(y|x)$ is the probability of a reference policy (often the referenced model) generating the same response. For each input data sample, there will be 2 representations, y_w and y_l, corresponding to the sample rated as accepted and rejected.

$$\mathcal{L}_{\text{DPO}}(\pi_\theta; \pi_{\text{ref}}) = -\mathbb{E}\left[\log \sigma\left(\beta \log \frac{r_\theta(y_w|x)}{r_\theta(y_l|x)}\right)\right] \tag{3}$$

In which,

$$r_\theta(y|x) = \frac{\pi_\theta(y|x)}{\pi_{\text{ref}}(y|x)}$$

Pairing Selection Mechanism: After ranking the candidate translations, we apply a pairing selection mechanism to create the reference-ranking dataset. For each source sentence in the seed data, there are 5 corresponding translations, and after ranking, we obtain up to 10 $\{y_{accepted} - y_{rejected}\}$ sentence pairs. In total, from the initial set of 2,000 sentences, we generate up to 20,000 samples for DPO training. To address the problem of close representation of preference ranking samples, we utilize the hybrid score of BLEU (Papineni et al., 2002) and METEOR (Lavie and Agarwal, 2007) for ranking the pairs as in Equation 4, wherein the $y_{accepted}$ sentence is considered as the reference sentence and the $y_{rejected}$ sentence serves as the candidate sentence. Finally, the top $- p\%$ sentence pairs with the highest hybrid scores are removed from the DPO training dataset.

$$\mathrm{h}(s) = 2 \cdot \frac{\tilde{\mathrm{bleu}}_s \cdot \tilde{\mathrm{meteor}}_s}{\tilde{\mathrm{bleu}}_s + \tilde{\mathrm{meteor}}_s} \quad (4)$$

In the dataset $\mathcal{D}$ consist of $\mathcal{N}$ pairs, for each pair of sentences $s = \{s_{ref}, s_{pred}\}$, in which s_{ref} is the reference sentence, s_{pred} is the prediction sentence.

$$\tilde{\mathrm{bleu}}_s = \frac{\mathrm{bleu}_s - \min_{i \in \mathcal{N}} \mathrm{bleu}_i}{\max_{i \in \mathcal{N}} \mathrm{bleu}_i - \min_{i \in \mathcal{N}} \mathrm{bleu}_i}$$
$$\tilde{\mathrm{meteor}}_s = \frac{\mathrm{meteor}_s - \min_{i \in \mathcal{N}} \mathrm{meteor}_i}{\max_{i \in \mathcal{N}} \mathrm{meteor}_i - \min_{i \in \mathcal{N}} \mathrm{meteor}_i}$$

4 Datasets

4.1 Available Datasets

We utilize three datasets for our experiments, as summarized in Table 1. The main dataset for supervised fine-tuning is the VOV corpus (Nguyen et al., 2022), which contains 135,164 training, 2,000 validation, and 2,000 test sentence pairs. Data are collected from bilingual news articles published by the Voice of Vietnam (VOV). In addition, we use 131,055 Khmer monolingual sentences from various sources to support language modeling tasks.

Dataset	Split	# samples	
		Vi	Km
Bilingual Dataset			
VOV	train	135,164	135,164
	valid	2,000	2,000
	test	2,000	2,000
Monolingual Dataset			
VOV	-	-	131,055
TED	-	353,251	1,066
QED	-	-	344

Table 1: Dataset statistics for our experiments

To ensure the quality of parallel dataset, we apply a two-stage filtering process: (1) Language Detection using Facebook AI's language identification tool (Bojanowski et al., 2017) to remove sentence pairs where the source is not Vietnamese or the target is not Khmer; and (2) Length Filtering, which eliminates pairs that are too short, too long, or have an abnormal length ratio between source and target sentences. After filtering, 135,164 high-quality sentence pairs are retained for fine-tuning.

For monolingual data, we assemble monolingual Vietnamese and Khmer sentences from multiple sources: the VOV datasets (Nguyen et al., 2022), which provides both a parallel Vi-Km subset and a standalone Khmer monolingual subset; the TED monolingual corpora for both languages (Reimers and Gurevych, 2020); and the QED Khmer monolingual corpus (Abdelali et al., 2014). To reduce noise, we apply a three-step pre-processing pipeline: (1) Remove Duplication using the MinHash algorithm; (2) Clean Special Characters such as HTML tags, non-printable ASCII characters, and emoji; and (3) Language Filtering to discard texts not in the target language.

After processing monolingual datasets, we accumulate a total of 760,066 monolingual samples, consisting of 490,426 Vietnamese sentences and 269,640 Khmer sentences.

4.2 Synthetic Data Generation

We propose a synthetic data generation framework, as in Figure 2, leveraging monolingual Vietnamese data, which is more abundant compared to Khmer, to construct high-quality synthetic parallel corpora. Our framework begins with selecting high-quality monolingual Vietnamese sentences using the TF-IDF score. These selected samples are then labeled through black-box knowledge distillation (Yang et al., 2024), in which `GPT-4o` serves as a teacher model to generate corresponding Khmer translations for training a student model. To validate and filter the synthetic data, we employ a back-translation strategy: instead of using the same teacher model, which could introduce circular reasoning due to shared architecture or biases (Wang and Sennrich, 2020), we use `Google Translate` as an independent translation system to back-translate the generated Khmer sentences into Vietnamese. To evaluate the fidelity of the back-translated sentences against the original Vietnamese inputs, we compute a hybrid quality score combining BLEU and METEOR (Equation 4).

4.2.1 Monolingual Data Selection

Among the available Vietnamese monolingual sources, the TED dataset is selected for its simple, domain-aligned sentences, which resemble those in our target training set. As shown in Figure 2, we sampled 30K TED sentences to balance quality and domain coverage, given that our SFT parallel data comprises around 130K examples. Although several selection methods exist—e.g., Cross-Entropy

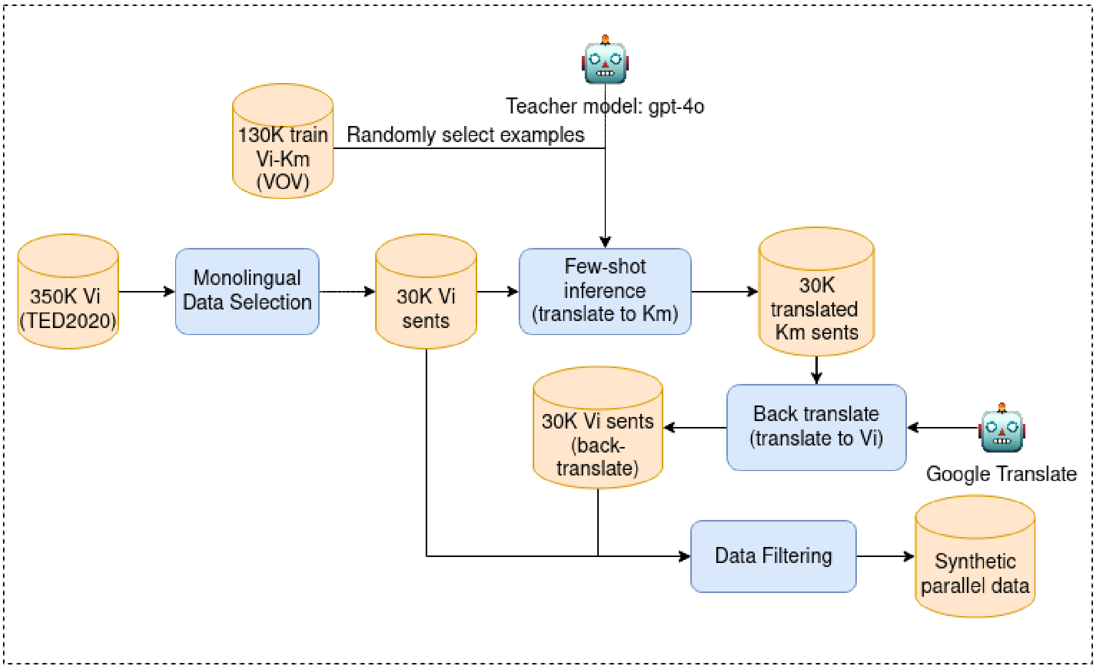

Figure 2: Synthetic data generation pipeline for **Vi-Km** translation task using VOV and TED datasets

Difference (Moore and Lewis, 2010), Feature Decay Algorithm (Poncelas et al., 2018), and TF-IDF; prior work (Silva et al., 2018) shows TF-IDF gains the best performance, and thus we adapt it in this stage. This proportion ensures a balance between input quality and domain alignment, helping to maintain strong performance on the in-domain VOV test set while preserving robustness under out-of-domain distribution shifts.

4.2.2 Synthetic Data Augmentation

We use the VOV bilingual dataset (Nguyen et al., 2022) as seed examples for few-shot prompting. For each of the 30K selected Vietnamese sentences, two randomly sampled examples are injected in the prompt to `GPT-4o` to guide diverse and high-quality Khmer translations.

After generating 30K synthetic Khmer sentences, we apply the back-translation method (Sennrich et al., 2016) to filter low-quality outputs. Specifically, each generated sentence is translated back into Vietnamese using `Google Translate` via the deep-translator library[3]. We compute a hybrid similarity score—defined as a weighted combination of BLEU (Papineni et al., 2002) and METEOR (Lavie and Agarwal, 2007) (see Equation 4)—between the original and back-translated Vietnamese sentences. Based on this score, we rank all sentence pairs in descending order and select the top $- p\%$ pairs. These original Vietnamese sentences are then matched with their corresponding Khmer outputs from `GPT-4o` to form the final synthetic parallel dataset. We choose not to use the back-translated Vietnamese sentences, as the translation quality tends to degrade after circular translation (Kementchedjhieva and Søgaard, 2023).

[3]https://github.com/nidhaloff/deep-translator

5 Experimental Results

5.1 Evaluating LoRA and AdaLoRA Efficiency on SFT

We evaluate two low-rank adapter methods, LoRA (Hu et al., 2022) and AdaLoRA (Zhang et al., 2023), with different configurations for SFT. In LoRA, the rank hyperparameter r controls the number of trainable parameters, balancing model capacity and efficiency. Meanwhile, in AdaLora, the $init\_r$ hyperparameter defines the initial low-rank dimensionality for each incremental adaptation matrix; conversely, $target\_r$ specifies the desired average rank across all adaptation matrices, directing the iterative pruning mechanism toward a predefined parameter budget and thereby balancing representational richness with computational and memory efficiency throughout training. Results comparing their effectiveness for our task are shown in Table 2.

LoRA consistently outperforms AdaLoRA on both BLEU and METEOR metrics, achieving higher scores of 58.67% and 54.18%, respectively.

Low-rank method	Configuration	%Trainable	Vi-Km BLEU/METEOR
AdaLoRA	$init\_r$: 12, $target\_r$: 8	0.8893	52.91/47.83
	$init\_r$: 20, $target\_r$: 16	1.4734	54.02/48.87
LoRA	r: 16	1.1820	**58.67/54.18**

Table 2: Low-rank adapters supervised fine-tuning results comparison

These results suggest that LoRA may be more effective in modeling the linguistic patterns, while AdaLoRA allows for dynamic rank adjustment and offers greater flexibility in parameter tuning, where its performance remained lower, even with an increased percentage of trainable parameters (1.4734%). This finding emphasizes LoRA's advantage in balancing model adaptation and translation quality. Based on this comparison, we adapt the LoRA configuration ($r = 16$, $\alpha = 32$) for the remainder of our experiments.

5.2 Main Results

Table 3 summarizes performance on both translation directions across our four training stages. The `SeaLLMs-v3-1.5B-Chat` baseline achieves quite low scores, indicating its limited zero-shot capability. In Stage 1, applying supervised fine-tuning (SFT) alone results in a substantial performance boost across both directions, improving BLEU by over 26% for Vi-Km and nearly as much for the reverse direction. Introducing SmCPT before SFT in Stage 2 further enhances translation quality, indicating that language-adaptive pretraining helps the model better align with the translation task. Stage 4 integrates DPO on top of SmCPT and SFT with synthetic preference ranking data. This approach achieves the highest performance, reaching BLEU/METEOR scores of 62.00/58.43(%) for Vi-Km and 57.12/54.61(%) for Km-Vi. These improvements confirm the effectiveness of preference-based alignment in refining model outputs.

5.3 Discussion

We conduct experiments on a diverse set of LLMs with few-shot prompts (0, 1, 2, 4, 8 shots), including GPT models, Gemini-Flash variants, and two commercial MT systems (`Google Translate`[4], and `Microsoft Translator`[5]). In parallel, we evaluate open-source PLM with Encoder-Decoder architecture baselines, including `mBART-50` (Liu et al., 2020), `NLLB-200-Distilled-600M` (Koishekenov et al., 2023).

Few-shot prompting systematically improves translation quality across GPT and Gemini models (see Table 4). Notably, the “mini” variant of these models remains substantially behind their "large" models by a clear margin. At the same time, commercial translation systems (`Google Translate`, `Microsoft Translator`) with direct translation rival many few-shot LLMs, indicating that traditional commercial MT systems remain strong baselines, especially for widely used translation tasks.

Besides, we also experiment with the efficiency of small language models over large-scale general language models on a specific task and language. Table 5 compares the fine-tuned performance of open-source NMT systems alongside previous Km-Vi translation studies. The most noticeable gains arise from fine-tuning open-source models on the domain-specific VOV dataset. Both `mBART-Large-50` and `NLLB-200-Distilled-600M` jump to over-50% BLEU after fine-tuning. One previous work, Pham Van and Le Thanh (2022) fine-tunes mBART50 on VOV and synthetic parallel data generated via back-translation and English-pivot augmentation, reporting a 54.50% BLEU on the Km-Vi VOV test set. Building on that, Quoc et al. (2023) introduce cosine-similarity-based data selection, synthetic candidate generation, and two-step filtering before fine-tuning, achieving 55.37% BLEU in the same translation direction. Meanwhile, Duc et al. (2025) focus on the Vi-Km task with a novel approach using joint-task training with the Question-Answering task, achieving 56.99% BLEU. Our fine-tuned model on the SeaLLMs-v3-1.5B-Chat baseline outperforms all these baselines—both closed-source and open-source—settings with zero-shot translation ability. These results emphasize the necessity of fine-tuning and domain-specific adaptation when leveraging LLMs for machine translation, especially in low-resource language scenarios.

6 Conclusion

This study tackles the challenge of machine translation for low-resource languages, focusing on the Vi-Km pair. We review existing approaches in low-resource machine translation, highlighting their limitations and identifying opportunities for improvement. To address these challenges, we

[4]`https://cloud.google.com/translate/docs`
[5]`https://www.microsoft.com/en-us/translator/`

Model/ Training Stage	Training Task	Augment Data	BLEU/METEOR Vi-Km	BLEU/METEOR Km-Vi
SeaLLMs-v3-1.5B-Chat	×	×	32.51/28.02	28.42/25.05
Stage 1	SFT	×	58.67/54.18	54.01/51.41
Stage 2	SmCPT, SFT	×	60.82/56.94	55.64/53.11
Stage 3	SmCPT, SFT	✓	61.13/57.44	56.89/54.12
Stage 4	SmCPT, SFT, DPO	✓	**62.00/58.43**	**57.12/55.61**

Table 3: Comparison of **Vi-Km and Km-Vi** translation performance at each stage

Model	# Few-shot samples	BLEU/METEOR Vi-Km	BLEU/METEOR Km-Vi
GPT-4o	0	49.60/51.54	52.80/51.82
	1	50.20/51.85	53.12/51.99
	2	50.85/52.02	53.55/52.21
	4	53.65/52.98	54.31/52.63
	8	54.45/54.03	55.07/54.34
GPT-4o-mini	0	50.12/44.90	49.34/45.64
	1	50.35/46.55	49.88/47.23
	2	51.06/47.34	51.32/49.22
	4	52.46/49.92	53.21/50.12
	8	53.02/49.33	54.37/52.21
GPT-3.5-turbo	0	43.74/16.30	44.32/20.21
	1	45.54/20.12	46.24/24.34
	2	47.66/23.45	47.85/25.21
	4	48.50/23.05	49.02/26.01
	8	49.00/23.98	49.32/28.88
Gemini-2.0-Flash	0	48.12/42.68	50.12/47.32
	1	49.23/43.54	51.11/48.23
	2	50.21/45.00	51.89/49.37
	4	51.68/48.21	52.32/50.75
	8	53.24/50.12	52.68/51.82
Gemini-1.5-Flash	0	39.72/16.24	40.44/18.21
	1	40.05/17.01	41.42/19.88
	2	40.24/17.44	41.78/20.21
	4	41.24/20.12	42.94/21.42
	8	42.73/22.01	43.94/23.67
Google Translate	×	49.74/50.37	51.21/52.23
Microsoft Translator	×	53.45/52.44	54.32/52.89
Ours (SeaLLMs-v3-1.5B)	×	**62.00/58.43**	**57.12/55.61**

Table 4: Results of different LLMs and software on the **Vi-Km and Km-Vi** translation task

Model/ Previous works	Fine-tune (VOV)	BLEU/METEOR Vi-Km	BLEU/METEOR Km-Vi
mBART-Large-50	×	37.67/8.94	31.41/5.32
	✓	50.56/48.90	52.84/49.98
NLLB-200-Distilled-600M	×	38.08/16.54	32.48/15.52
	✓	50.33/49.20	51.97/49.51
Pham Van and Le Thanh (2022) (mBART-50)	✓	-/-	54.50/-
Quoc et al. (2023) (mBART-50)	✓	-/-	55.37/-
Duc et al. (2025) (Sealion-3B (Singapore, 2024))	✓	56.99/-	-/-
SeaLLMs-v3-1.5B	×	32.51/28.02	28.42/25.05
Ours (SeaLLMs-v3-1.5B)	✓	**62.00/58.43**	**57.12/55.61**

Table 5: Results of different open-source models and previous works on the **Vi-Km and Km-Vi** translation task

propose a lightweight, modular training pipeline that leverages small-scale continual pretraining on monolingual data, supervised fine-tuning, and Direct preference optimization, with synthetic data augmentation generated by high-performance LLMs.

The empirical experiment is built upon the `SeaLLMs-v3-1.5B-Chat` model, a small multilingual ASEAN PLM. By leveraging the LoRA adapter, we efficiently multi-stage train the model and achieved the highest performance, with 62.00/58.43 and 57.12/55.61 (%) (BLEU/METEOR) on the evaluation set. These results outperform commercial translation systems (Google Translate, Microsoft Translator), LLMs such as GPT-4o and Gemini models, and also previous works (Pham Van and Le Thanh, 2022; Quoc et al., 2023; Duc et al., 2025).

These results highlight the limitations of general-purpose LLMs in handling low-resource language pairs and demonstrate the effectiveness of task-specific adaptation. The proposed training framework, which focuses on a cost-effective, lightweight, and comprehensive method, provides a promising and scalable direction for future research on machine translation in low-resource languages.

For future direction, there still exists room for expanding the quality of research, not only for Vietnamese-Khmer pair, but also for other regional low-resource machine translation. In particular, future work may investigate adaptive tokenization strategies that better capture the linguistic properties of languages, as well as the use of cross-lingual transfer learning from typologically or geographically related languages. Furthermore, integrating evaluation methods that combine automatic metrics with human-in-the-loop assessment would provide a more holistic measure of translation quality and better guide model development.

Acknowledgments

This research is funded by Hanoi University of Science and Technology (HUST) under Grant no T2024-PC-041.

References

Ahmed Abdelali, Francisco Guzman, Hassan Sajjad, and Stephan Vogel. 2014. The AMARA corpus: Building parallel language resources for the educational domain. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 1856–1862, Reykjavik, Iceland. European Language Resources Association (ELRA).

Abhaya Agarwal and Alon Lavie. 2008. Meteor, m-bleu and m-ter: Evaluation metrics for high-correlation with human rankings of machine translation output. In *WMT@ACL*.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

Tho Tran Duc, Huy Nguyen Trong, and Huong Le Thanh. 2025. Improving quality of vietnamese to khmer neural machine translation using multi-stage fine-tuning strategy. In *Information and Communication Technology*, pages 69–79, Singapore. Springer Nature Singapore.

Sergey Edunov, Myle Ott, Marc'Aurelio Ranzato, and Michael Auli. 2020. On the evaluation of machine translation systems trained with back-translation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2836–2846, Online. Association for Computational Linguistics.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*.

Yova Kementchedjhieva and Anders Søgaard. 2023. Grammatical error correction through round-trip machine translation. In *Findings of the Association for Computational Linguistics: EACL 2023*, pages 2208–2215, Dubrovnik, Croatia. Association for Computational Linguistics.

Yunsu Kim, Petre Petrov, Pavel Petrushkov, Shahram Khadivi, and Hermann Ney. 2019. Pivot-based transfer learning for neural machine translation between non-English languages. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 866–876, Hong Kong, China. Association for Computational Linguistics.

Yeskendir Koishekenov, Alexandre Berard, and Vassilina Nikoulina. 2023. Memory-efficient NLLB-200: Language-specific expert pruning of a massively multilingual machine translation model. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3567–3585, Toronto, Canada. Association for Computational Linguistics.

Zhanibek Kozhirbayev. 2024. Enhancing neural machine translation with fine-tuned mbart50 pre-trained model: An examination with low-resource translation pairs. *Ingénierie des systèmes d information*, 29:831–838.

Sneha Kudugunta, Isaac Caswell, Biao Zhang, Xavier Garcia, Derrick Xin, Aditya Kusupati, Romi Stella, Ankur Bapna, and Orhan Firat. 2023. Madlad-400: A multilingual and document-level large audited dataset. In *Advances in Neural Information Processing Systems*, volume 36, pages 67284–67296. Curran Associates, Inc.

Alon Lavie and Abhaya Agarwal. 2007. Meteor: An automatic metric for mt evaluation with high levels of correlation with human judgments. In *WMT@ACL*.

Yinhan Liu, Jiatao Gu, Naman Goyal, Xian Li, Sergey Edunov, Marjan Ghazvininejad, Mike Lewis, and Luke Zettlemoyer. 2020. Multilingual denoising pre-training for neural machine translation. *Transactions of the Association for Computational Linguistics*, 8:726–742.

Zihan Liu, Genta Indra Winata, and Pascale Fung. 2021. Continual mixed-language pre-training for extremely low-resource neural machine translation. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 2706–2718, Online. Association for Computational Linguistics.

Mieradilijiang Maimaiti, Yang Liu, Huanbo Luan, Zegao Pan, and Maosong Sun. 2021. Improving data augmentation for low-resource nmt guided by pos-tagging and paraphrase embedding. *ACM Trans. Asian Low-Resour. Lang. Inf. Process.*, 20(6).

Robert C. Moore and William Lewis. 2010. Intelligent selection of language model training data. In *Proceedings of the ACL 2010 Conference Short Papers*, pages 220–224, Uppsala, Sweden. Association for Computational Linguistics.

Yasmin Moslem, Rejwanul Haque, John D. Kelleher, and Andy Way. 2023. Adaptive machine translation with large language models. In *Proceedings of the 24th Annual Conference of the European Association for Machine Translation*, pages 227–237, Tampere, Finland. European Association for Machine Translation.

Linh The Nguyen and Dat Quoc Nguyen. 2025. Pre-training of foundation adapters for LLM fine-tuning. In *The Fourth Blogpost Track at ICLR 2025*.

Vinh Van Nguyen, Ha Nguyen, Huong Thanh Le, Thai Phuong Nguyen, Tan Van Bui, Luan Nghia Pham, Anh Tuan Phan, Cong Hoang-Minh Nguyen, Viet Hong Tran, and Anh Huu Tran. 2022. KC4MT: A high-quality corpus for multilingual machine translation. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 5494–5502, Marseille, France. European Language Resources Association.

Xuan-Phi Nguyen, Wenxuan Zhang, Xin Li, Mahani Aljunied, Zhiqiang Hu, Chenhui Shen, Yew Ken Chia, Xingxuan Li, Jianyu Wang, Qingyu Tan, Liying Cheng, Guanzheng Chen, Yue Deng, Sen Yang, Chaoqun Liu, Hang Zhang, and Lidong Bing. 2024. SeaLLMs - large language models for Southeast Asia. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, pages 294–304, Bangkok, Thailand. Association for Computational Linguistics.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Hanh Pham Van and Huong Le Thanh. 2022. Improving khmer-vietnamese machine translation with data augmentation methods. In *Proceedings of the 11th International Symposium on Information and Communication Technology*, SoICT '22, page 276–282, New York, NY, USA. Association for Computing Machinery.

Alberto Poncelas, Gideon Maillette de Buy Wenniger, and Andy Way. 2018. Data selection with feature decay algorithms using an approximated target side. In *Proceedings of the 15th International Conference on Spoken Language Translation*, pages 173–180, Brussels. International Conference on Spoken Language Translation.

Thai Nguyen Quoc, Huong Le Thanh, and Hanh Pham Van. 2023. Khmer-vietnamese neural machine translation improvement using data augmentation strategies. *Informatica*, 47(3):349–359. Bibliografija: str. 345-347.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model. In *Thirty-seventh Conference on Neural Information Processing Systems*.

Nils Reimers and Iryna Gurevych. 2020. 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.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Improving neural machine translation models with monolingual data. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 86–96, Berlin, Germany. Association for Computational Linguistics.

Catarina Cruz Silva, Chao-Hong Liu, Alberto Poncelas, and Andy Way. 2018. Extracting in-domain training corpora for neural machine translation using data selection methods. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 224–231, Brussels, Belgium. Association for Computational Linguistics.

AI Singapore. 2024. Sea-lion (southeast asian languages in one network): A family of large language models for southeast asia. `https://github.com/aisingapore/sealion`.

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 *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 6000–6010, Red Hook, NY, USA. Curran Associates Inc.

Chaojun Wang and Rico Sennrich. 2020. On exposure bias, hallucination and domain shift in neural machine translation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3544–3552, Online. Association for Computational Linguistics.

Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. 2021. mT5: A massively multilingual pre-trained text-to-text transformer. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 483–498, Online. Association for Computational Linguistics.

Yuzi Yan, Yibo Miao, Jialian Li, YipinZhang, Jian Xie, Zhijie Deng, and Dong Yan. 2025. 3d-properties: Identifying challenges in DPO and charting a path forward. In *The Thirteenth International Conference on Learning Representations*.

Chuanpeng Yang, Yao Zhu, Wang Lu, Yidong Wang, Qian Chen, Chenlong Gao, Bingjie Yan, and Yiqiang Chen. 2024. Survey on knowledge distillation for large language models: Methods, evaluation, and application. *ACM Trans. Intell. Syst. Technol.* Just Accepted.

Weizhe Yuan, Richard Yuanzhe Pang, Kyunghyun Cho, Xian Li, Sainbayar Sukhbaatar, Jing Xu, and Jason Weston. 2024. Self-rewarding language models. In *Proceedings of the 41st International Conference on Machine Learning*, ICML'24. JMLR.org.

Qingru Zhang, Minshuo Chen, Alexander Bukharin, Pengcheng He, Yu Cheng, Weizhu Chen, and

Tuo Zhao. 2023. Adaptive budget allocation for parameter-efficient fine-tuning. In *The Eleventh International Conference on Learning Representations*.

Renrui Zhang, Jiaming Han, Chris Liu, Aojun Zhou, Pan Lu, Yu Qiao, Hongsheng Li, and Peng Gao. 2024. LLaMA-adapter: Efficient fine-tuning of large language models with zero-initialized attention. In *The Twelfth International Conference on Learning Representations*.

Wenhao Zhu, Hongyi Liu, Qingxiu Dong, Jingjing Xu, Shujian Huang, Lingpeng Kong, Jiajun Chen, and Lei Li. 2024. Multilingual machine translation with large language models: Empirical results and analysis. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 2765–2781, Mexico City, Mexico. Association for Computational Linguistics.

A Training Configuration

We conduct the model training and evaluation experiments on a single GPU RTX 4090. The total time for the whole training process took over 50 hours. For loading and training the models, we utilize the modules: Transformers and TRL[6] from HuggingFace. The training process consists of three main stages: Small-scale Continual Pretraining (SmCPT), Supervised Fine-tuning (SFT), and Direct Preference Optimization (DPO). The hyperparameters used in each task are shown in Table 6.

Hyperparameter	SmCPT	SFT	DPO
number of epochs	1	2	2
batch size	8	4	4
learning rate	1e-4	1e-5	5e-6
warmup ratio	0.1	0.1	0.1
compute data type	bfloat16	bfloat16	bfloat16
optimizer	adamw_8bit	adamw_8bit	adamw_8bit
lr scheduler type	cosine	cosine	cosine
beta (β)	-	-	0.3

Table 6: Training Configuration

Small-scale Continual Pre-training (SmCPT) is conducted with a relatively high learning rate (1e-4) and a larger batch size to quickly adapt the pretrained language model to domain-specific data while preserving general knowledge. Since the objective is to adapt to a specific task and domain rather than full-scale pre-training, only one epoch is used to prevent overfitting. In the Supervised Fine-tuning (SFT) phase, a lower learning rate (1e-5) ensures stable refinement without erasing prior knowledge, while the batch size is reduced to 4 to ensure stable updates. Since DPO training requires fine-grained updates, an even smaller learning rate (5e-6) is used to prevent drastic deviations from prior alignment. Across all training stages, computations are conducted in bfloat16 precision, which accelerates training compared to fp32 while preventing overflow when using fp16.

[6]`https://huggingface.co/docs/trl/index`

B Ablation Studies

B.1 Synthetic Data Filtering Effect

The overall pipeline of data augmentation is described in Section 4.2. Firstly, 30,000 Vietnamese sentences were selected from the seed dataset of TED (Reimers and Gurevych, 2020) by TF-IDF score. Then, we used `GPT-4o` model through the OpenAI API[7] with two-shot learning, taking a random combination from the training VOV dataset, to generate the translated Khmer settings. Two main hyperparameters need to be controlled during using LLM generation: *temperature* and *top_p*. In which *temperature* is a hyperparameter that controls the randomness of language model output, meaning a high temperature produces more unpredictable and creative results, and vice versa. Meanwhile, *top_p* is also used for controlling the randomness of the language model; it sets a threshold probability and selects the top tokens whose cumulative probability exceeds the threshold, which helps deliver more diverse and interesting output. For our task, we set *temperature*=0.1 and *top_p*=0.99, which is the best configuration for the translation task as in the experiment by Moslem et al. (2023).

The generated Khmer sentences are then back-translated to Vietnamese by using `Google Translate`. These sentences are compared to the original Vietnamese sentences using the hybrid score of BLEU and METEOR (see Equation 4). We conduct the experiment that combines the $top-p\%$ score pairs and VOV dataset to self-supervised fine-tune the model from the SmCPT checkpoint, as outlined in Section 3.2. As indicated from Table 7, just selecting the 90% highest score from augmented data shows the best performance in both BLEU and METEOR scores, with 61.13 and 57.44 respectively.

B.2 Preference-ranking Pairing Data Selection

We experiment with two strategies to create a DPO training dataset from the maximum of $20,000$ pref-

[7]`https://platform.openai.com/docs`

Threshold	Augmented	Vi-Km
($top - p\%$)	**Data Size**	**BLEU/METEOR**
100%	$30,000$	$60.88/57.07$
95%	$28,500$	$60.51/56.68$
$\mathbf{90\%}$	$\mathbf{27,000}$	$\mathbf{61.13/57.44}$
85%	$25,500$	$60.77/56.88$

Table 7: Results of top-$p\%$ augmented data selection on the performance of fine-tuning

erence ranking pairs generated as described in Section 3.2.3. The strategy (1) is pairing the highest-ranking denoted as y_1 with the lower-ranking sentences denoted as $\{y_2, y_3, y_4, y_5\}$. The strategy (2) uses all of $20,000$ combinations, then computes the score of each pair (see Equation 4) to select $top - p\%$ highest score pairs as the DPO dataset.

No	Ranking pair		Threshold	Vi-Km
	Accepted	**Rejected**	($top - p\%$)	**BLEU/METEOR**
(1)	y_1	$\{y_5\}$	-	$61.85/58.13$
	y_1	$\{y_4, y_5\}$		$61.91/58.40$
	y_1	$\{y_3, y_4, y_5\}$		$61.74/58.06$
	y_1	$\{y_2, y_3, y_4, y_5\}$		$61.63/58.11$
(2)	All combinations		100%	$61.86/58.24$
			95%	$61.88/58.41$
			90%	$62.00/\mathbf{58.43}$
			85%	$61.85/58.28$
			80%	$\mathbf{62.04}/58.41$
			75%	$61.70/58.05$

Table 8: Results of different DPO dataset selection strategies on the performance

Compared to previous stage results (see Table 7), DPO training significantly improves the performances in all cases (see Table 8). These demonstrate the importance of filtering low-quality ranking pairs to achieve optimal model parameters. At this stage, we select the model with a $top - p\%$ of 90%, achieving a BLEU score of 62.00 (slightly lower than the 80% $top - p\%$ model by 0.04). However, it peaks at a METEOR score of 58.43. Based on the conclusion from Agarwal and Lavie (2008) and our study, we conclude that METEOR offers a more rigorous and accurate evaluation compared to BLEU. This supports our decision to prioritize METEOR as a primary metric when determining the best-performing model.

B.3 Impact of Hyperparameter-beta on DPO Performance

We set up an experiment to study the effect of hyperparameter-β in DPO loss (see Equation 3). We use 2,000 preference ranking pairs that map the highest and the lowest ranking translated sentence ranked by `GPT-4o`. We adjusted the value of β from 0.05 to 0.5, and the results on BLEU and METEOR metrics corresponding to each value of β are (see Figure 3).

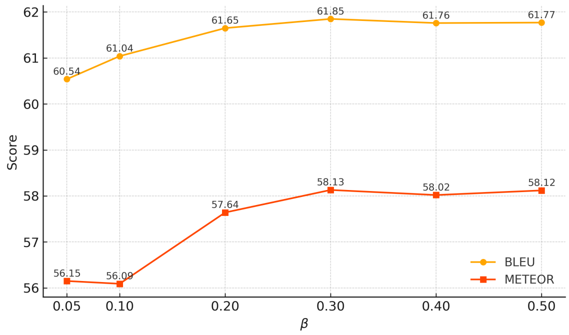

Figure 3: Impact of hyperparameter β on DPO performance

It can be observed that at β=0.3, the model achieves its best performance, with BLEU and METEOR scores of 61.85 and 58.13, respectively. Adjusting β significantly impacts the model's performance, making the selection of an appropriate β crucial for the specific downstream task. When β is smaller, the penalty term in the loss function is reduced, leading to less distinct separation between win and loss sequences. This setting is more suitable for tasks requiring creative and diverse outputs. On the other hand, for tasks where accurate outputs aligned with the input sequence are critical, choosing a larger β ensures stricter penalization and more precise outputs.

B.4 Impact of Dataset Size on SFT Performance

We provide an analysis of the impact of training data size on the SFT performance. For the experiment, we fine-tune the Adalora adapter (Zhang et al., 2023) on the training set. As shown in the results (Figure 4), there is a consistent improvement in model performance as the size of the training data increases, indicating a strong correlation between data quantity and translation quality. The BLEU score is 36.32% for a training size of 25K samples and progressively increases to 52.91% when fine-tuning on the full dataset. Similarly, the METEOR scores show a steady rise, from 31.68% for 25K samples to 47.83% for the full dataset. These findings emphasize the critical role of data size in supervised fine-tuning models, indicating that increased data may lead to better model perfor-

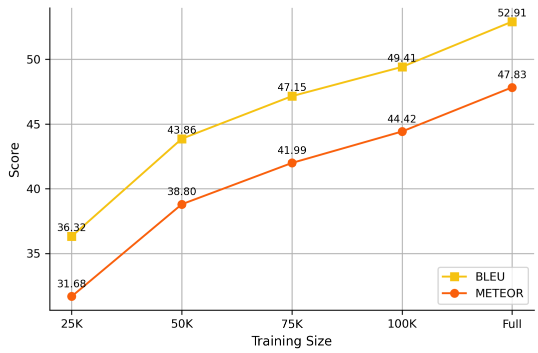

Figure 4: Impact of data training size on the SFT performance

mance, particularly for low-resource languages.

C Prompt Design

##Instruction: Translate from {source lang} to {target lang}

#Input ({source lang}):
{Input sentence}

#Output ({target lang}):

Table 9: Prompt for fine-tuning translation task

Table 9 presents the template used for fine-tuning the translation model. The instruction specifies the translation task, with inputs labeled as "Input (Vietnamese)" and outputs labeled as "Output (Khmer)".

##Instruction: Here are some examples of {source lang}-to-{target lang} translations:

#Example 1:
{source lang}: {Input sample sentence 1}
{target lang}: {Output sample sentence 1}

#Example 2:
{source lang}: {Input sample sentence 2}
{target lang}: {Output sample sentence 2}

##Your task: Now, translate the following {source lang} sentence into {target lang}:
{source lang}: {Input sentence}
{target lang}:

Table 10: Prompt for generating Khmer translation on random two-shot examples

A two-shot learning prompt is used to generate synthetic Khmer translations with the `GPT-4o` model (see Table 10), including an instruction that provides examples of Vietnamese-to-Khmer translations and demonstrates the expected format and style of translation. After the examples, the model is assigned to translate a new Vietnamese sentence into Khmer. This prompt format incorporates in-context learning by showing examples and immediately asking the model to perform the task. In the experiment, based on the strength of the teacher model, only two examples are provided to balance the token length and the cost.

##Instruction: Your task is to rank the following {target lang} translation candidates based on the **accuracy** and **naturalness** relative to the source {source lang} sentence and the golden reference {target lang} translation.

#Original {source lang}:
{Input sentence}

#Reference {target lang}:
{Golden translation sentence}

#Candidates:
1. {Candidate translation 1}
2. {Candidate translation 2}
3. {Candidate translation 3}
4. {Candidate translation 4}
5. {Candidate translation 5}

##Your task: Ranking the candidates from the best to the worst in the format:
{candidate_number_1, candidate_number_2, ..., candidate_number_5}.

Table 11: Prompt for LLM translation ranking

Table 11 outlines the prompt used for ranking Khmer translation candidates, the idea of which is based on LLM-as-a-judge (Yuan et al., 2024). The instruction asks the model to rank the given candidate translations based on **accuracy** and **naturalness** relative to the original Vietnamese sentence and a ‘golden’ reference Khmer translation.