Low-Resource Language Expansion and Translation Capacity Enhancement for LLM: A Study on the Uyghur

Kaiwen Lu[1,2,3], **Yating Yang**[1,2,3*], **Fengyi Yang**[1,2,3], **Rui Dong**[1,2,3], **Bo Ma**[1,2,3], **Ahtamjan Ahmat**[1,2,3], **Abibilla Atawulla**[1,2,3], **Lei Wang**[1,2,3], **Xi Zhou**[1,2,3]

[1]Xinjiang Technical Institute of Physics & Chemistry, Chinese Academy of Sciences, Urumqi, China

[2]University of Chinese Academy of Sciences, Beijing, China

[3]Xinjiang Laboratory of Minority Speech and Language Information Processing, Urumqi, China

{lukaiwen20,aihetamujiangaihemaiti20}@mails.ucas.ac.cn
{yangyt,yangfy,dongrui,mabo,aibibula,wanglei,zhouxi}@ms.xjb.ac.cn

Abstract

Although large language models have significantly advanced natural language generation, their potential in low-resource machine translation has not yet been fully explored, especially for languages that translation models have not been trained on. In this study, we provide a detailed demonstration of how to efficiently expand low-resource languages for large language models and significantly enhance the model's translation ability, using Uyghur as an example. The process involves four stages: collecting and pre-processing monolingual data, conducting continuous pre-training with extensive monolingual data, fine-tuning with less parallel corpora using translation supervision, and proposing a direct preference optimization based on translation self-evolution (DPOSE) on this basis. Extensive experiments have shown that our strategy effectively expands the low-resource languages supported by large language models and significantly enhances the model's translation ability in Uyghur with less parallel data. Our research provides detailed insights for expanding other low-resource languages into large language models.

1 Introduction

The emergence of large language models(LLMs) has brought a new paradigm to the field of natural language processing(NLP) (Achiam et al., 2023; Touvron et al., 2023a; Chowdhery et al., 2023; Touvron et al., 2023b; Anil et al., 2023; Xu et al., 2024). LLMs have gained remarkable understanding and the extraordinary ability to generate human-like text due to their enormous parameter size and massive training data. From the initial Transformer to the later BERT series(Kenton and Toutanova, 2019; Zhang et al., 2019; Liu et al., 2020), and GPT series(Radford, 2018; Radford et al., 2019; Brown, 2020; Ouyang et al., 2022; Achiam et al., 2023), the models' parameters have become increasingly larger, and their understanding and generation capabilities have also become stronger. Until today, representative LLMs such as BLOOM(and: et al., 2023) and GPT4(OpenAI et al., 2024) have become well-known in the field of NLP. These LLMs have demonstrated outstanding performance across a range of benchmark tests and exhibit translation capabilities that are comparable to the best machine translation models in scenarios involving high-resource languages.(Hendy et al., 2023; Zhu et al., 2024a). Compared to rich languages, LLMs are not sufficiently trained on low-resource languages. Limited by the initial training corpus of the model itself, LLMs often only perform well in some resource-rich languages, while they perform poorly or even fail to understand most low-resource languages(Mao and Yu, 2024; Merx et al., 2024). This is mainly reflected in the fact that these low-resource languages have a tiny proportion of the model's training data and a sparse vocabulary(Ebrahimi and Kann, 2021). As illustrated in

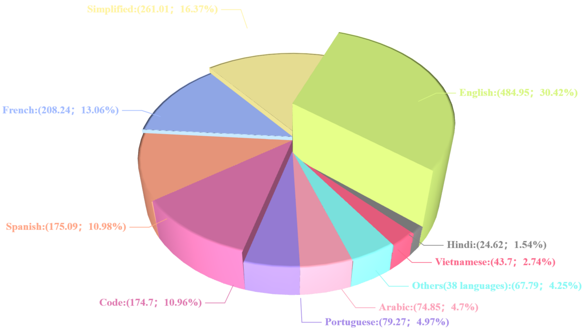


Figure 1: An example of visualizing the distribution of training data languages in the BLOOM. This example provides statistics on the byte size of each language in the 1.61TB of training data used by BLOOM, with the unit of measurement being billions.

[*] Corresponding author

Proceedings of the 31st International Conference on Computational Linguistics, pages 8360–8373
January 19–24, 2025. ©2025 Association for Computational Linguistics

Figure 1, the nine most prevalent languages within the BLOOM training dataset constitute 95.75% of the dataset's total volume, whereas the remaining 38 languages collectively account for only 4.25%, with the least represented language comprising a mere 0.16MB. This inherent limitation of big models in supporting low-resource languages severely hinders the development of these languages.

The purpose of this paper is to study the language expansion and enhancement of translation capabilities for low-resource languages on LLM. We use Uyghur as an example, which is an agglutinative language primarily used in the Xinjiang Uyghur Autonomous Region of China, with about 13 million speakers. Although there are many speakers, recent research on the language has been scarce, and the corpus available for collection is very limited. Based on our limited exploration and testing (see section 5.2), there is currently no LLM that can effectively understand and translate Uyghur, which severely hinders the development of the language.

Building upon an LLM that has been enhanced with Chinese capabilities, Chinese-LLaMA2-7B(Cui et al., 2024), which is an upgrade from LLaMA2(Touvron et al., 2023b), we have comprehensively realized the expansion of the LLM's Uyghur capabilities and enhanced their translation abilities. This was achieved through a series of steps, including data collection and processing, pre-training, translation instruction fine-tuning, and direct preference optimization(Rafailov et al., 2023) based on translation self-evolution (**DPOSE**). The proposed method has been validated on the CCMT-UC(CCMT, 2024)[1] benchmark machine translation test set for Uyghur. The contributions of this paper can be summarized as follows:

- We propose an effective data collection and pre-processing method for low-resource languages and have published our training data distribution, which is currently rarely mentioned in related work. Subsequent experiments validate our strategy.
- We introduce a DPOSE, a simple yet effective training strategy that significantly enhances the translation capability of low-resource languages and alleviates the translation off-target issue in LLM.
- We have been instrumental in expanding the LLM to include Uyghur and enhancing its translation capabilities, thereby offering valuable insights for extending similar improvements to other low-resource languages within the LLMs.

2 Related Work

2.1 LLMs for Low-resource Language

For languages underrepresented in LLMs, performance can be effectively enhanced through fine-tuning and the application of prompting methods. (Huang et al., 2023) introduced a universal template prompt designed to systematically stimulate cross-language and logical reasoning skills, thereby enhancing the multilingual capabilities of LLMs. This approach is referred to as Cross-linguistic Thought Prompts (XLT). (Lai et al., 2023) evaluated ChatGPT comprehensively across seven distinct tasks, spanning 37 languages, with a spectrum of resources ranging from high to extremely low. The assessment revealed that ChatGPT's performance was notably subpar for low-resource languages. (Guo et al., 2024) constructed a textbook-like corpus for LLMs by referencing the way humans learn, and improved the models' understanding of low-resource languages through learning from textbooks. (Yamaguchi et al., 2024) systematically investigated existing cross-lingual vocabulary adaptation methods and demonstrated that models pre-trained on more balanced multilingual data can achieve comparable downstream performance to the original models.

The other extreme case is how LLMs can understand languages never trained. To equip LLMs with the capability to process unseen languages, the most straightforward strategy is to expand the vocabulary and augment the training data for such languages. (Cui et al., 2024) improved the coding efficiency of LLaMA and its semantic understanding of Chinese by expanding the existing vocabulary to additional Chinese tokens. (Kim et al., 2024) proposed an efficient vocabulary expansion method, encompassing parameter freezing and subword initialization. (Fujii et al., 2024) extended the vocabulary of LLaMA2 to include Japanese characters and conducted continual pre-training on a large Japanese web corpus. (Gao et al., 2024) proposed XConST, a cross-linguistic consistency regularization technique, to reduce the representation disparity among various languages and to

[1] The corpus is the most extensive publicly accessible Uyghur translation resource, comprising a parallel dataset of 170,000 Uyghur-Chinese text pairs.

Language	Code	Family	Script	Size
Uyghur	uy	Turkic	Arabic	14487M
Kazakh	kk	Turkic	Arabic	228M
Tajik	tg	Turkic	Slavic	579M
Turkish	tr	Turkic	Latin	2052M
English	en	Germanic	Latin	3099M
Arabic	ar	Arabic	Arabic	2916M
Chinese	zh	Chinese	Chinese	6639M

Table 1: Statistical distribution of pre-processed pre-trained data. This table counts the number of tokens for each language, using a unit of millions.

improve the zero-shot translation capabilities of LLMs.

2.2 LLMs for Machine Translation

The advent of LLMs has concurrently introduced novel opportunities for machine translation. (Zhu et al., 2024b) proposed cross-lingual examples that offer enhanced task guidance for low-resource translation. (Mao and Yu, 2024) enhances the performance of LLMs in zero-shot translation by constructing translation instructions and cross-lingual alignment instructions. (Zhang et al., 2023b) compared the performance of three methods: zero-shot prompting, few-shot learning, and fine-tuning, validating them on both sentence-level and document-level translation tasks. (Zhang et al., 2023a) transferred language generation and instruction tracking abilities from English to other languages by employing interactive translation tasks and has developed a model translation LLM named BayLing. (Yin et al., 2024a) proposed a straightforward yet effective methodology for data collection, which employs bilingual dictionaries to generate a dataset. Following the fine-tuning of the collected data, the LLM demonstrated significant performance improvements, particularly in tasks involving word sense disambiguation and specialized terminology translation.

3 Datasets Collection and Pre-processing

Our Uyghur data is sourced directly from Uyghur websites and the CC-100 dataset[2]. We have implemented a set of heuristic pre-processing techniques for all data, which include rule-based filtering, and data deduplication. (1) **Rule-based Filtering**: includes keyword filtering, abnormal character filtering, and ad filtering. (2) **Data Deduplication**: employed the xorbits[3] tool, which leverages the MinHash-LSH(Zhu et al., 2016) algorithm for efficient corpus deduplication promptly.

Name	Perplexity↓
w/o pre-processing	343.75
basic pre-processing	11.53
+ high-quality data	10.03
+ comprehensive pre-processing	8.46

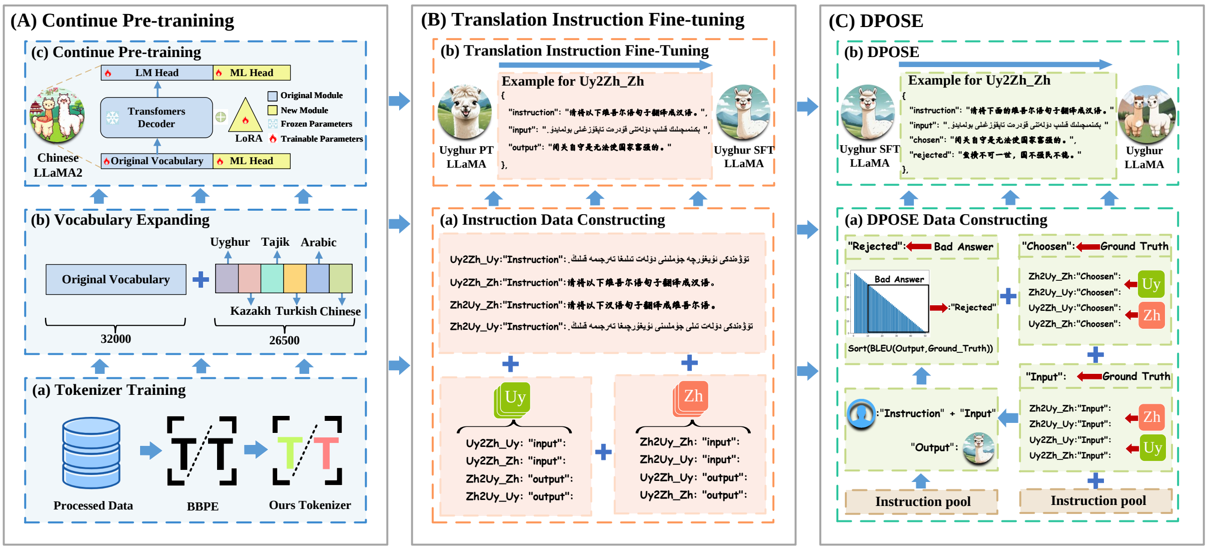
Table 2: Statistical distribution of pre-processed pre-trained data. This table counts the number of tokens for each language, using a unit of millions.

The data we have collected totals 30B, with a language distribution as shown in Table 1. This table reveals that, alongside Uyghur, six additional languages have been incorporated into the dataset. Empirically, this approach serves two key objectives: firstly, to preserve the model's initial understanding of other languages as effectively as possible, thereby mitigating the risk of catastrophic forgetting; and secondly, to enrich the model's representation, given that the data provided by Uyghur is not sufficiently comprehensive. By including languages similar to Uyghur, the model's capacity for knowledge transfer is significantly enhanced. Subsequently, we segmented the data into high-quality and low-quality categories. High-quality data comprises sentences that are reliable in origin, coherent, and rich in knowledge, due to its resource limitations, this part of data accounts for about 0.4389% of the total data. The remaining data is considered low-quality data.

To validate the effectiveness of our data processing strategy, we pre-trained models in four different scenarios to verify our strategy, namely (1) no pre-processing at all, (2) only basic pre-processing (keyword filtering and abnormal character filtering), (3) adding high-quality data to the basic pre-processing, and (4) adding high-quality data to comprehensive pre-processing. We used perplexity to evaluate the models' outputs, and the calculation of perplexity(jel, 1977), specifically, we calculate the cross-entropy loss by comparing the model output with real samples and then take the base-10 exponential function of the result to represent perplexity. As detailed in Table 2, the model's performance clearly shows the advantages of employing effective data pre-processing techniques and ensuring high data quality. To illustrate the differences

[2] https://data.statmt.org/cc-100/

[3] https://github.com/xorbitsai/xorbits



```
{
  "instruction": "请将以下维吾尔语句子翻译成汉语。",
  "input": "پكىنمىچىلىك قىلىپ دۆلەتنى قۇدرەت تاپقۇزغىلى بولمايدۇ.",
  "output": "闭关自守是无法使国家富强的。"
},
```

```
Uy2Zh_Uy:"Instruction":تۆۋەندىكى ئۇيغۇرچە جۈملىنى دۆلەت تىلىغا تەرجىمە قىلىڭ.
Uy2Zh_Zh:"Instruction":请将以下维吾尔语句子翻译成汉语。
Zh2Uy_Zh:"Instruction":请将以下汉语句子翻译成维吾尔语。
Zh2Uy_Uy:"Instruction":تۆۋەندىكى دۆلەت تىلى جۈملىنى ئۇيغۇرچىغا تەرجىمە قىلىڭ.
```

```
Uy2Zh_Uy: "input":
Uy2Zh_Zh: "input":
Zh2Uy_Zh: "output":
Zh2Uy_Uy: "output":
```

```
Zh2Uy_Zh: "input":
Zh2Uy_Uy: "input":
Uy2Zh_Uy: "output":
Uy2Zh_Zh: "output":
```

```
{
  "instruction": "请将下面的维吾尔语句子翻译成汉语。"
  "input": "پكىنمىچىلىك قىلىپ دۆلەتنى قۇدرەت تاپقۇزغىلى بولمايدۇ. "
  "chosen": "闭关自守是无法使国家富强的。",
  "rejected": "蛮横不可一世，国不强民不稳。"
},
```

```
Zh2Uy_Zh:"Choosen":
Zh2Uy_Uy:"Choosen":
Uy2Zh_Uy:"Choosen":
Uy2Zh_Zh:"Choosen":
```

```
Zh2Uy_Zh:"Input":
Zh2Uy_Uy:"Input":
Uy2Zh_Uy:"Input":
Uy2Zh_Zh:"Input":
```

Figure 2: Illustrates of training process flowchart. Blue arrows denote sequential relationships, blue plus signs represent concatenation and combination, and red arrows signify assignment.

intuitively, we present examples of Strategy 1 and 4 in Appendix A.

4 Method

In this section, we first briefly introduce our methodology for pre-training and the supervised fine-tuning of translation. We then present our method DPOSE. Our training process is shown in Figure 2.

4.1 Pre-training

Pre-training is the foundational and critical step that determines a model's ability to fluently output a language, consuming the most resources and time. We continued the pre-training of the Chinese LLaMA2 model on a standard causal language modeling (CLM) task. Given an input token sequence $x = (x_0, x_1, x_2, \ldots)$, we trained the model to predict the next token x_i in a self-regressive manner. Mathematically, the objective is to minimize the negative log-likelihood:

$$\mathcal{L}_{PT}(\Theta) = \mathbb{E}\left[-\sum_i \log p(x_i | x_0, x_1, \ldots, x_{i-1} | \Theta) \right], \quad (1)$$

where, Θ represents the model parameters, x_i is the token to be predicted, and $x_0, x_1, \ldots, x_{i-1}$ constitute the context.

As illustrated in Figure 2(A), like many language expansion efforts, our pre-training strategy is characterized by three primary aspects. Initially, we leveraged Byte Pair Encoding (BPE)(Gage, 1994) to retrain a specialized tokenizer, which was informed by the six languages detailed in Table 1. The second key detail is that we set the total number of tokens for the tokenizer to 30K. Subsequently, after applying the tokenizer, we obtained a new vocabulary. After comparing and de-duplicating this with the original vocabulary, we obtained a new vocabulary of 26.5K tokens. Finally, we concatenated and merged the two vocabularies. The third critical aspect involves retraining the model using the LoRA approach. Specifically, we fully trained the model's embedding layer and LM head. The model's attention layers and MLP layers were maintained in a frozen state, with only a minimal additional set of parameters trained for this part. Detailed parameter settings are described in Section 5.1.

4.2 Translations Instruction Fine-tuning

Fine-tuning of language models has become a standard practice in the field of LLM-based translation (Jiao et al., 2023; Xu et al., 2024; Yin et al., 2024b). This process involves adapting a pre-trained language model to a specific task or domain by training it on a smaller, more specialized dataset. The goal is to improve the model's performance on the target task by enhancing its understanding of the domain-specific vocabulary, syntax, and context. Through fine-tuning, LLM-based translation systems can achieve higher accuracy and better contextual understanding, making them more effective for practical applications.

The instruction-following data is constructed from the set $(S = S_r \cup S_c)$. Generally, each instance consists of an "instruction" i describing the task the model should perform (e.g., "Translate the sentences from Uyghur to Chinese."), an "input" x indicating the source sentence, and a corresponding output y indicating the answer to the instruction, i.e., the target sentence. The language models are optimized by minimizing the negative log-likelihood of the output y:

$$\mathcal{L}_{TIF}(\Theta) = -\sum_{(x,y)\in S} \frac{1}{|y|} \sum_{i} |y| \log p(y_i|c,x;\Theta), \quad (2)$$

where Θ represents the trainable model parameters. It is important to note that to ensure the translation capability of our model is as robust as possible across various prompting scenarios, we have implemented four distinct modes in the design of our instructions, as illustrated in Figure 2(B). Specifically, we have used two languages as prompts in each direction, A sample of the final data for translation instructions is shown in Appendix B.1.

4.3 DPOSE

Although LLMs can acquire the ability to fluently output language through extensive unsupervised learning, the challenge of precisely controlling their behavior persists due to the completely unsupervised nature of their training. To address this issue, reinforcement learning from human feedback (RLHF) is commonly utilized. However, this process is complex and frequently unstable. Consequently, we have adopted a direct preference optimization (DPO) method as a benchmark in this paper, which is simpler to implement and more effective than RLHF, while requiring fewer computational resources.

DPO is commonly used in the training of question-answering and security domains to generate more useful and secure answers. The standard DPO data format includes a question and two or more alternative answers, which are classified into *"chosen"* and *"rejected"* categories. During training, the model is trained to favor *"chosen"* answers, thereby aligning its preferences with those of humans (Rafailov et al., 2023). Motivated by this, we have explored applying DPO to the field of machine translation. As illustrated in Figure 2(C), we have treated the *"instruction"* and *"input"* (source language) from the SFT phase as the *"instruction"* and *"input"* for DPO, with the corresponding Ground Truth (target language) of the *"input"* serving as the *"chosen"* for DPO. Subsequently, we have used the responses of the SFT-trained model to questions as *"rejected"* to complete the construction of the DPO data for self-evolution in translation (DPOSE). It is important to note that upon examining the translation outputs of the SFT model, we observed that certain responses were of a higher quality than the reference translation. Consequently, we computed the SacreBLEU(Post, 2018) scores for all responses and designated the lowest 70% of samples as "rejected" for subsequent refinement. Meanwhile, the top 30% of high-quality responses were excluded from further participation in the DPOSE process. The optimization goal is as follows:

$$\mathcal{L}_{\text{DPOSE}}\left(\pi_\theta; \pi_{\text{ref}}\right) = -\mathbb{E}_{(s,t_c,t_r)}\left[\log \sigma\left(\log R\right)\right], \quad (3)$$

where, π represents the parameter distribution of the language model, π_θ represents the current model distribution, which is initialized as π_{SFT}, π_{ref} represents the model distribution under an ideal state, s represents the source language to be translated, t_c represents the chosen answer, t_r represents the rejected answer, σ is a logical function, and the representation of R is as follows:

$$R = \beta \frac{\pi_\theta\left(t_c \mid x\right)}{\pi_{\text{ref}}\left(t_c \mid x\right)} - \beta \log \frac{\pi_\theta\left(t_r \mid x\right)}{\pi_{\text{ref}}\left(t_r \mid x\right)}, \quad (4)$$

where β is a parameter controlling the deviation, which is set to 0.1. Applying DPOSE transforms our optimization objective into encouraging the model to generate answers that are congruent with the preferences of human translators. A sample of the final DPOSE data is shown in Appendix B.2.

5 Experiments

5.1 Basic Details

Baseline Settings We have constructed our model on the foundation of Chinese-LLaMA2-7B(Cui et al., 2024), a model that was derived by continuously pre-training and fine-tuning the LLaMA2-7B base model with supplementary Chinese data. To enhance training efficiency, we conduct full-scale training on the model's embedding layer and LM-head layer, employing the low-rank adaptation (LoRA)(Hu et al., 2022) fine-tuning for the remainder of the components.

Parameter Settings The parameter settings are detailed in Table 4. For the continuous pre-training and translation supervision fine-tuning phases, we

Model	Uy→Zh		Zh→Uy	
	BLEU	COMET	BLEU	COMET
LLaMA2-7B (Touvron et al., 2023b)	0.1	-138.7	0	-179.9
ChatGLM3-6B (Du et al., 2022)	0.3	-135.5	0	-168.7
Chinese LLaMA2-7B (Cui et al., 2024)	0.4	-133.3	0	-167.9
Qwen-7B (Bai et al., 2023)	0.6	-113.0	0	-155.0
Qwen1.5-14B (Team, 2024)	4.5	-50.7	0.1	-120.8
Baichuan2-13B (Yang et al., 2023)	6.0	-30.3	0.1	-142.0
GPT-3.5 (Ouyang et al., 2022)	10.9	-14.6	0.9	-84.2
GPT-4 (OpenAI et al., 2024)	**19.5**	**23.2**	**1.9**	**-49.7**

Table 3: Statistics on the performance of LLMs for translation between Uyghur and Chinese.

Settings	Pre-training	SFT	DPOSE
Training data	30B	29M	67M
Batch size	1,024	512	128
Peak learning rate	1e-4	1e-4	5e-5
Max sequence length	512	512	1024
LoRA rank	8	8	8
LoRA alpha	32	32	16
Trainable params	6.06%	6.22%	3.14%
Accuracy	fp16	fp16	bf16

Table 4: Parameter settings for the continuous pre-training and fine-tuning of translation instructions.

adopted the parameter configurations from the Chinese LLaMA2. Given the sensitivity of the DPOSE phase parameters, we opted for the values presented above after a series of limited tests. The training above was all conducted under the Llama-Factory[4] framework.

Training costs All of our experiments were conducted on Nvidia A100 (80G) GPUs. During the pre-training phase, we used 16 cards and completed this phase in 20 days. For the instruction fine-tuning phase, we used 8 cards and finished the training in 1 day. In the DPOSE phase, we also used 8 cards and completed the training in 7 days.

Evaluation Metrics We employ BLEU and COMET(Rei et al., 2020) as evaluation metrics to evaluate the performance of our models. For BLEU, we utilize the SacreBLEU implementation, which standardizes tokenization and enhances reproducibility. Unlike the BLEU metric, which relies on the overlap of n-grams between a machine-generated translation and a reference translation, COMET models are trained on a comprehensive dataset that includes human translations and assessments of their quality. This dataset is leveraged to predict translation quality, and the source text is also considered. This method allows COMET to offer a more comprehensive evaluation that encompasses fluency, adequacy, and the preservation of meaning. We utilize the latest model, Unbabel/wmt22-cometda[5], for our evaluation.

5.2 Uyghur Translation Testing for LLMs

A natural question to ask is whether existing LLMs with representative capabilities can effectively translate the Uyghur. Consequently, we evaluated these models using the CCMT-UC test set in both Uyghur→Chinese and Chinese→Uyghur translation scenarios. The models tested include: (1) **GPT-4** (OpenAI et al., 2024) (2) **GPT-3.5** (Ouyang et al., 2022) (3) **Baichuan2-13B** (Yang et al., 2023) (4) **Qwen 1.5-14B** (Team, 2024) (5) **Qwen-7B** (Bai et al., 2023) (6) **Chinese LLaMA2-7B** (Cui et al., 2024) (7) **ChatGLM3-6B** (Du et al., 2022) (8) **LLaMA 2-7B** (Touvron et al., 2023b). The results of the testing are shown in Table 3. The majority of existing LLMs are incapable of generating Uyghur texts (Zh→Uy) and exhibit limited comprehension of Uyghur texts (Uy→Zh). Notably, GPT-4, GPT-3.5, and Baichuan-2 have demonstrated relatively strong performance in the Uy→Zh direction, yet there remains room for enhancement in the Zh→Uy direction. This suggests that these models can grasp the semantics of Uyghur texts but are unable to produce reliable Uyghur texts. This observation underscores the importance of our investigation into the Uyghur language. Additionally, we have conducted a detailed test of the Uy→Zh results, shown in Appendix C.1.

[4] https://github.com/hiyouga/LLaMA-Factory

Model	Uy→Zh_Zh		Uy→Zh_Uy		Zh→Uy_Uy		Zh→Uy_Zh	
	BLEU	COMET	BLEU	COMET	BLEU	COMET	BLEU	COMET
LLaMA2-7B (Touvron et al., 2023b)	0.1	-138.7	0	-165.4	0	-179.9	0	-179.8
ChatGLM3-6B (Du et al., 2022)	0.3	-135.5	0	-157.8	0	-168.7	0	-169.7
Chinese LLaMA2-7B (Cui et al., 2024)	0.4	-133.3	0	-164.8	0	-167.9	0	-169.9
Qwen-7B (Bai et al., 2023)	0.6	-113	0	-175.8	0	-155	0	-155.0
Qwen1.5-14B (Team, 2024)	4.5	-50.7	0.3	-158.6	0.1	-120.8	0.1	-120.7
Baichuan2-13B (Yang et al., 2023)	6	-30.3	3.9	-72.7	0.1	-142	0.1	-142.0
GPT-3.5 (Ouyang et al., 2022)	10.9	-14.6	8.2	-26.9	0.9	-84.2	0.4	-129.8
GPT-4 (OpenAI et al., 2024)	19.5	23.2	12.3	3.22	1.9	-49.7	0.8	-92.13
Uyghur LLaMA PT	23.6	27.7	19.5	21.5	7.2	23.8	5.1	0.7
Uyghur LLaMA SFT	33.2	41.8	31.6	34.5	11.1	39.2	9.1	34.7
Uyghur LLaMA DPOSE	**33.7**	**43.6**	**32.1**	**39.3**	**12.1**	**45.0**	**10.3**	**36.7**

Table 5: The statistical results of the test on the impact of the language of the prompt on the translation between Chinese and Uyghur. The notation Uy→Zh_Zh indicates that Uyghur is translated into Chinese, with Chinese prompt, and so on.

Model	Uy→Zh		Zh→Uy	
	BLEU	COMET	BLEU	COMET
Chinese LLaMA2	0.4	-133.3	0	-167.9
Uyghur LLaMA PT	23.6	27.7	7.2	23.8
Uyghur LLaMA SFT	33.2	41.8	11.1	39.2
Uyghur LLaMA DPOSE	**33.7**	**43.6**	**12.1**	**45.0**

Table 6: Uyghur LLaMA training strategy effectiveness validation score statistics.

5.3 Uyghur LLaMA

To validate the effectiveness of our language expansion strategy and translation capability enhancement method, we evaluated our pre-trained model, SFT model, and DPOSE model on the Uy⇔ Zh two translation directions, as illustrated in Table 6. Our approach significantly enhanced the model's comprehension and translation proficiency in Uyghur. Specifically, the DPOSE method, compared to our baseline, saw improvements in BLEU scores of 33.3 and 11.1 in the two translation directions, with COMET improving by 176.9 and 205.1 respectively.

5.4 Efficacy of Prompt

Unlike the paradigm of traditional generative tasks, the output of LLMs is significantly influenced by the design of prompts. Effective prompts can markedly enhance the model's performance (Gao et al., 2024). Furthermore, for translation models that can generate output in multiple languages, the language of the prompt also has a notable impact. Consequently, we compared the language of the prompt for Uyghur LLaMA, as illustrated in Table 5. Prompts can have varying effects, with those that align with the target language proving more effective for translation. Current LLMs rely on an auto-regressive mechanism that generates output token by token. This reliance on previous outputs means that using prompts that match the target language can offer the model more accurate supervision signals to guide the output, thereby enhancing the likelihood of producing the correct word order in the target language. We also found that the BLEU score of Uyghur are relatively lower, this is because Uyghur is a agglutinative language, which consists of a word stem and affixes, meaning that if the model outputs the correct word stem but the affix is wrong, it will still be judged as a wrong token.

5.5 Evaluation of Off-Target Translations

Off-target translation, i.e., translating into the incorrect target language, is a major factor contributing to the poor quality of LLMs translations. To assess the accuracy of our model in the translation between Chinese and Uyghur, we employed Fasttext(Bojanowski et al., 2017) to analyze the model's output. Subsequently, we used the translation language hit rate P_{hr} to evaluate the model's capability of accurately translating into the correct target language. The definition of P_{hr} is as follows:

$$P_{hr} = (1 - n_{on-target}/n_{sentences}), \qquad (5)$$

where $n_{on-target}$ refer to total number of on-target

[5] https://huggingface.co/Unbabel/wmt22-comet-da

Model	Uy→Zh_Zh	Uy→Zh_Uy	Zh→Uy_Uy	Zh→Uy_Zh
LLaMA2-7B (Touvron et al., 2023b)	40.5	0.2	5.6	3.7
ChatGLM3-6B (Du et al., 2022)	71.5	0.6	0	0
Chinese LLaMA2-7B (Cui et al., 2024)	68.7	6.1	0.5	1.8
Qwen-7B (Bai et al., 2023)	91.2	0.6	12.4	25.8
Qwen1.5-14B (Team, 2024)	98.2	3.8	64.4	15.3
Baichuan2-13B (Yang et al., 2023)	94.5	35.9	77.7	20.6
GPT-3.5 (Ouyang et al., 2022)	95.2	84.2	88.3	64.1
GPT-4 (OpenAI et al., 2024)	97.9	95.3	84.7	75.3
Uyghur LLaMA PT	97.5	95.4	99.9	99.8
Uyghur LLaMA SFT	98	96.5	100	100
Uyghur LLaMA DPOSE	**98.5**	**98.3**	**100**	**100**

Table 7: Language hit rate(%) in four translation scenarios.

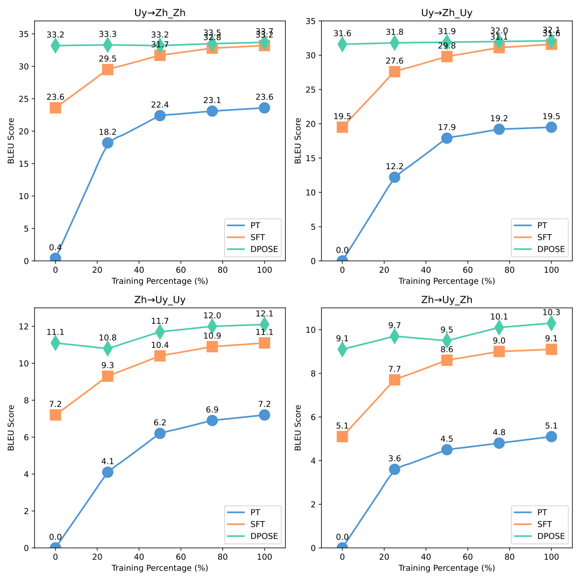


Figure 3: Learning curves of training in three stages for four types of translation scenarios.

sentences and $n_{sentences}$ means total number of sentences. The outcomes are presented in Table 7.

5.6 Learning Curves

As seen in Figure 3, a higher language hit rate signifies greater translation capability. Our three-stage model demonstrates a commendable performance in this regard. In addition, we have found that different prompts significantly impact the language hit rate of the model in the same translation direction, as exemplified by models other than our own and GPT. This phenomenon can be attributed to the models can partially understand the meaning of Uyghur content but their inability to produce output in Uyghur.

To illustrate the learning curves in our three stages across four translation scenarios, we have sampled an average of five checkpoints at each stage of the training process as examples. We evaluate the model's performance at each checkpoint by computing the BLEU scores for the four scenarios. The learning curves are depicted in Figure 3. It is evident that the model's performance has significantly improved in the PT and SFT stages. Conversely, the DPOSE stage exhibits its main improvements in the two scenarios where Uyghur is the target language. Furthermore, the training stability of DPOSE also displays some fluctuations, which are attributed to the instability of the DPO algorithm.

6 Conclusion

In this paper, we provide a detailed account of how to effectively expand the language varieties of LLM with less data, including data collection and pre-processing, data distribution, continuous pre-training, and SFT. On this basis, we propose the DPOSE method, and our multiple experiments show that our method effectively enhances the low-resource language translation ability of the model. In our experimental analysis, we observed that: (1) the application of effective pre-processing strategies and the utilization of high-quality data significantly enhance the pre-training process, particularly for languages with limited resources; (2) prompts that are aligned with the target language prove to be more effective in the context of LLMs for translation tasks; and (3) improvements made for low-resource languages can bring more improvements than those for high-resource languages.

In future work, we are interested in exploring that how to enhance cross-linguistic knowledge transfer within LLMs. We believe that continuous pre-training is sufficient to enable models to acquire the ability to fluently output language, and that LLMs possess well generalization capabilities.

Limitations

Despite notable contributions, this study has certain limitations. Firstly, while we conducted tests of Uyghur language capabilities on eight representative LLMs, this is still not comprehensive. Secondly, the phenomenon of hallucination in translation was observed in LLMs, but it was not extensively explored. Future research should delve deeper into how to mitigate the occurrence of this phenomenon. Lastly, this paper mainly focuses on the performance of Uyghur machine translation. Still, there is also an opportunity to explore performance under different tasks, such as the applicability of our methods in areas like question answering and literary creation. Due to computational resource constraints, we will comprehensively explore these areas in the future.

Acknowledgments

This project is mainly supported by the Tianshan Talent Training Program under Grant 2022TSYCCX0059, 2023TSYCCX0041 and 2023TSYCCX0044, in part by the Outstanding Member Program of the Youth Innovation Promotion Association of Chinese Academy of Sciences under Grant Y2021112, Y2023118 and 2021436, in part by the Natural Science Foundation of Xinjiang Uyghur Autonomous Region under Grant 2022D01D04 and 2024D01D29, in part by the ZF pre-research project, Grant 30601010202, in part by the National Natural Science Foundation of China, Grant 62406317.

References

1977. Perplexity—a measure of the difficulty of speech recognition tasks. *The Journal of the Acoustical Society of America*, 62(S1):S63–S63.

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

BigScience Workshop and:, Teven Le Scao, Angela Fan, Christopher Akiki, Ellie Pavlick, Suzana Ilić, Daniel Hesslow, Roman Castagné, and et al. 2023. Bloom: A 176b-parameter open-access multilingual language model. *Preprint*, arXiv:2211.05100.

Rohan Anil, Andrew M Dai, Orhan Firat, Melvin Johnson, Dmitry Lepikhin, Alexandre Passos, Siamak Shakeri, Emanuel Taropa, Paige Bailey, Zhifeng Chen, et al. 2023. Palm 2 technical report. *arXiv preprint arXiv:2305.10403*.

Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, and et al. 2023. Qwen technical report. *Preprint*, arXiv:2309.16609.

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.

Tom B Brown. 2020. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*.

CCMT. 2024. The 20th china conference on machine translation. *CCMT blog*.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, et al. 2023. Palm: Scaling language modeling with pathways. *Journal of Machine Learning Research*, 24(240):1–113.

Yiming Cui, Ziqing Yang, and Xin Yao. 2024. Efficient and effective text encoding for chinese llama and alpaca. *Preprint*, arXiv:2304.08177.

Zhengxiao Du, Yujie Qian, Xiao Liu, Ming Ding, Jiezhong Qiu, Zhilin Yang, and Jie Tang. 2022. GLM: General language model pretraining with autoregressive blank infilling. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 320–335, Dublin, Ireland. Association for Computational Linguistics.

Abteen Ebrahimi and Katharina Kann. 2021. How to adapt your pretrained multilingual model to 1600 languages. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 4555–4567, Online. Association for Computational Linguistics.

Kazuki Fujii, Taishi Nakamura, Mengsay Loem, Hiroki Iida, Masanari Ohi, Kakeru Hattori, Hirai Shota, Sakae Mizuki, Rio Yokota, and Naoaki Okazaki. 2024. Continual pre-training for cross-lingual LLM adaptation: Enhancing japanese language capabilities. In *First Conference on Language Modeling*.

Philip Gage. 1994. A new algorithm for data compression. *The C Users Journal*, 12(2):23–38.

Pengzhi Gao, Zhongjun He, Hua Wu, and Haifeng Wang. 2024. Towards boosting many-to-many multilingual machine translation with large language models. *Preprint*, arXiv:2401.05861.

Ping Guo, Yubing Ren, Yue Hu, Yunpeng Li, Jiarui Zhang, Xingsheng Zhang, and Heyan Huang. 2024. Teaching large language models to translate on low-resource languages with textbook prompting. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 15685–15697, Torino, Italia. ELRA and ICCL.

Amr Hendy, Mohamed Abdelrehim, Amr Sharaf, Vikas Raunak, Mohamed Gabr, Hitokazu Matsushita, Young Jin Kim, Mohamed Afify, and Hany Hassan Awadalla. 2023. How good are gpt models at machine translation? a comprehensive evaluation. *Preprint*, arXiv:2302.09210.

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*.

Haoyang Huang, Tianyi Tang, Dongdong Zhang, Xin Zhao, Ting Song, Yan Xia, and Furu Wei. 2023. Not all languages are created equal in LLMs: Improving multilingual capability by cross-lingual-thought prompting. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 12365–12394, Singapore. Association for Computational Linguistics.

Wenxiang Jiao, Jen-tse Huang, Wenxuan Wang, Zhiwei He, Tian Liang, Xing Wang, Shuming Shi, and Zhaopeng Tu. 2023. ParroT: Translating during chat using large language models tuned with human translation and feedback. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 15009–15020, Singapore. Association for Computational Linguistics.

Jacob Devlin Ming-Wei Chang Kenton and Lee Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of naacL-HLT*, volume 1, page 2.

Seungduk Kim, Seungtaek Choi, and Myeongho Jeong. 2024. Efficient and effective vocabulary expansion towards multilingual large language models. *Preprint*, arXiv:2402.14714.

Viet Lai, Nghia Ngo, Amir Pouran Ben Veyseh, Hieu Man, Franck Dernoncourt, Trung Bui, and Thien Nguyen. 2023. ChatGPT beyond English: Towards a comprehensive evaluation of large language models in multilingual learning. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 13171–13189, Singapore. Association for Computational Linguistics.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Ro{bert}a: A robustly optimized {bert} pretraining approach.

Zhuoyuan Mao and Yen Yu. 2024. Tuning LLMs with contrastive alignment instructions for machine translation in unseen, low-resource languages. In *Proceedings of the Seventh Workshop on Technologies for Machine Translation of Low-Resource Languages (LoResMT 2024)*, pages 1–25, Bangkok, Thailand. Association for Computational Linguistics.

Raphaël Merx, Aso Mahmudi, Katrina Langford, Leo Alberto de Araujo, and Ekaterina Vylomova. 2024. Low-resource machine translation through retrieval-augmented LLM prompting: A study on the Mambai language. In *Proceedings of the 2nd Workshop on Resources and Technologies for Indigenous, Endangered and Lesser-resourced Languages in Eurasia (EURALI) @ LREC-COLING 2024*, pages 1–11, Torino, Italia. ELRA and ICCL.

OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, and et al. 2024. Gpt-4 technical report. *Preprint*, arXiv:2303.08774.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems*, volume 35, pages 27730–27744. Curran Associates, Inc.

Matt Post. 2018. A call for clarity in reporting BLEU scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 186–191, Brussels, Belgium. Association for Computational Linguistics.

A Radford. 2018. Improving language understanding by generative pre-training.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

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 *Advances in Neural Information Processing Systems*, volume 36, pages 53728–53741. Curran Associates, Inc.

Ricardo Rei, Craig Stewart, Ana C Farinha, and Alon Lavie. 2020. COMET: A neural framework for MT evaluation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2685–2702, Online. Association for Computational Linguistics.

Qwen Team. 2024. Introducing qwen1.5.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. 2023a. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023b. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Haoran Xu, Young Jin Kim, Amr Sharaf, and Hany Hassan Awadalla. 2024. A paradigm shift in machine translation: Boosting translation performance of large language models. In *The Twelfth International Conference on Learning Representations*.

Atsuki Yamaguchi, Aline Villavicencio, and Nikolaos Aletras. 2024. An empirical study on cross-lingual vocabulary adaptation for efficient language model inference. *Preprint*, arXiv:2402.10712.

Aiyuan Yang, Bin Xiao, Bingning Wang, Borong Zhang, Ce Bian, Chao Yin, Chenxu Lv, Da Pan, Dian Wang, and et al. 2023. Baichuan 2: Open large-scale language models. *Preprint*, arXiv:2309.10305.

Yongjing Yin, Jiali Zeng, Yafu Li, Fandong Meng, and Yue Zhang. 2024a. Lexmatcher: Dictionary-centric data collection for llm-based machine translation. *CoRR*, abs/2406.01441.

Yongjing Yin, Jiali Zeng, Yafu Li, Fandong Meng, and Yue Zhang. 2024b. Lexmatcher: Dictionary-centric data collection for llm-based machine translation. *Preprint*, arXiv:2406.01441.

Shaolei Zhang, Qingkai Fang, Zhuocheng Zhang, Zhengrui Ma, Yan Zhou, Langlin Huang, Mengyu Bu, Shangtong Gui, Yunji Chen, Xilin Chen, and Yang Feng. 2023a. Bayling: Bridging cross-lingual alignment and instruction following through interactive translation for large language models. *Preprint*, arXiv:2306.10968.

Xuan Zhang, Navid Rajabi, Kevin Duh, and Philipp Koehn. 2023b. Machine translation with large language models: Prompting, few-shot learning, and fine-tuning with QLoRA. In *Proceedings of the Eighth Conference on Machine Translation*, pages 468–481, Singapore. Association for Computational Linguistics.

Zhengyan Zhang, Xu Han, Zhiyuan Liu, Xin Jiang, Maosong Sun, and Qun Liu. 2019. ERNIE: Enhanced language representation with informative entities. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1441–1451, Florence, Italy. Association for Computational Linguistics.

Erkang Zhu, Fatemeh Nargesian, Ken Q Pu, and Renée J Miller. 2016. Lsh ensemble: internet-scale domain search. *Proceedings of the VLDB Endowment*, 9(12):1185–1196.

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

Wenhao Zhu, Hongyi Liu, Qingxiu Dong, Jingjing Xu, Shujian Huang, Lingpeng Kong, Jiajun Chen, and Lei Li. 2024b. 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 Pre-processing strategies comparison

As illustrated in Figure 4, our pre-processing strategy significantly enhanced performance, underscoring the critical role of an effective pre-processing strategy in establishing the model's foundational capabilities.

B Data Example

In this section, we demonstrate the data format we used during the SFT and DPOSE training.

B.1 SFT Data Example

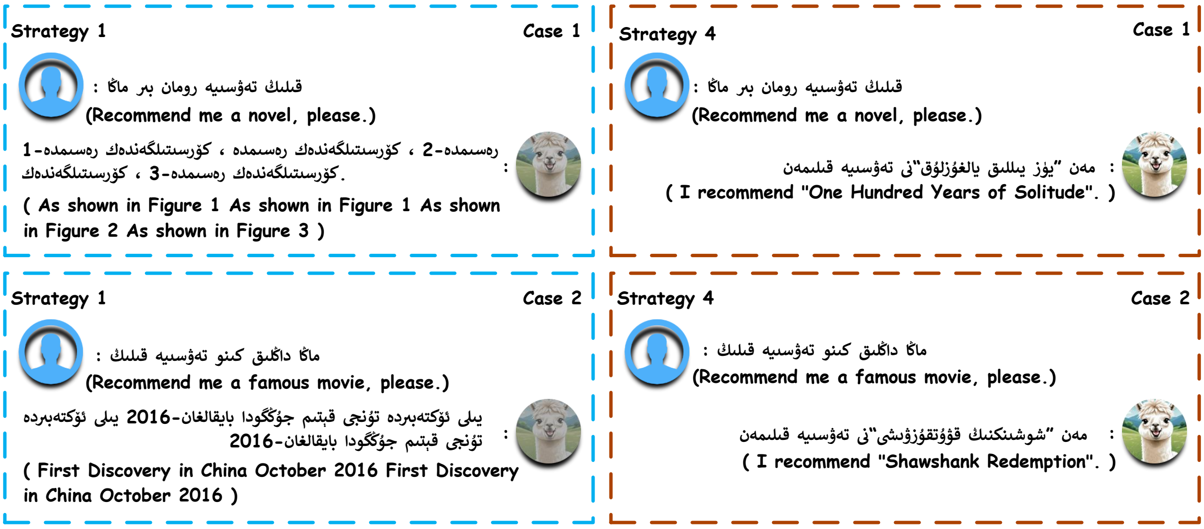
As shown in Figure 5, we utilized parallel corpora of Uyghur and Chinese to construct four types of SFT data. The figure illustrates the four ways in which we processed a pair of parallel sample.

B.2 DPOSE Data Example

As shown in Figure 6, we first constructed question-answer pairs using parallel corpora of Uyghur and Chinese, and then used two prompting methods in each translation direction to have the SFT model answer the questions. After that, we calculated the BLEU score for the answers. The 70% with the lowest scores were used as $"rejected"$, and the corresponding $"instructions"$ and $"inputs"$ were retained. The reference answers were kept as $"chosen"$.

C Supplement to Experimental Results

In this section, we have provided a comprehensive supplement and explanation of the experimental results in the article.



Strategy 1 Case 1

قىلىڭ تەۋسىيە رومان بىر ماڭا :
(Recommend me a novel, please.)

: رەسىمدە-2 ، كۆرسىتىلگەندەك رەسىمدە ، كۆرسىتىلگەندەك رەسىمدە-1
كۆرسىتىلگەندەك. رەسىمدە-3 ، كۆرسىتىلگەندەك
(As shown in Figure 1 As shown in Figure 1 As shown in Figure 2 As shown in Figure 3)

Strategy 4 Case 1

قىلىڭ تەۋسىيە رومان بىر ماڭا :
(Recommend me a novel, please.)

: مەن ”يۈز يىللىق يالغۇزلۇق“نى تەۋسىيە قىلىمەن
(I recommend "One Hundred Years of Solitude".)

Strategy 1 Case 2

ماڭا داڭلىق كىنو تەۋسىيە قىلىڭ :
(Recommend me a famous movie, please.)

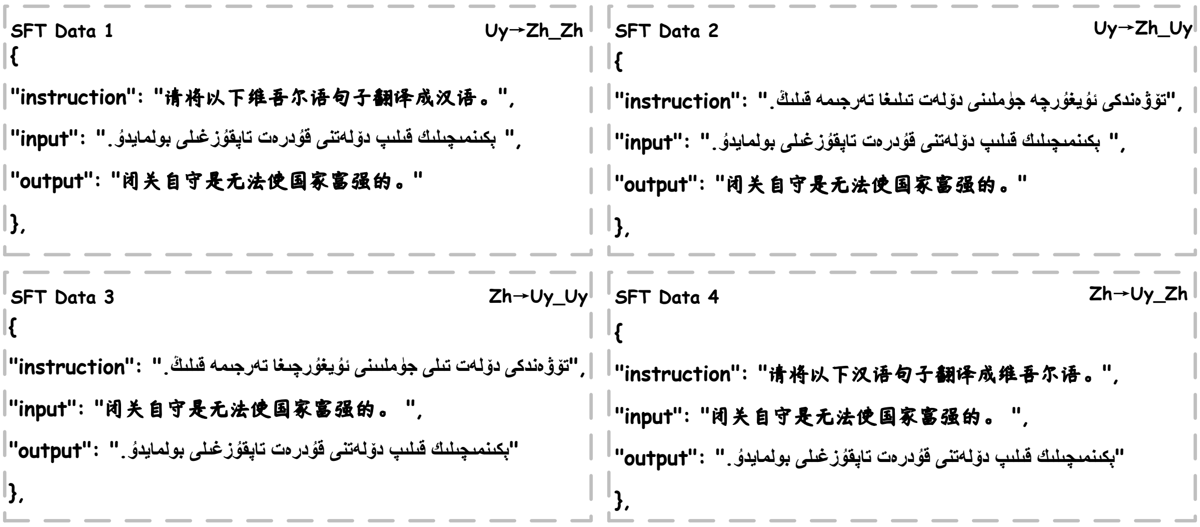
: يىلى ئۆكتەبىردە تۇنجى قېتىم جۇڭگودا بايقالغان-2016 يىلى ئۆكتەبىردە
تۇنجى قېتىم جۇڭگودا بايقالغان-2016
(First Discovery in China October 2016 First Discovery in China October 2016)

Strategy 4 Case 2

ماڭا داڭلىق كىنو تەۋسىيە قىلىڭ :
(Recommend me a famous movie, please.)

: مەن ”شوشىنكنىڭ قۇتقۇزۇلۇشى“نى تەۋسىيە قىلىمەن
(I recommend "Shawshank Redemption".)

Figure 4: A graphical representation comparing the effectiveness of various data pre-processing strategies reveals that Strategy 1 corresponds to the performance of models that undergo pre-training without any preliminary data manipulation. Conversely, Strategy 4 denotes the performance of models that are pre-trained following meticulous data refinement and the incorporation of high-quality datasets.



SFT Data 1 Uy→Zh_Zh
```
{
"instruction": "请将以下维吾尔语句子翻译成汉语。",
"input": "بېكىنمىچىلىك قىلىپ دۆلەتنى قۇدرەت تاپقۇزغىلى بولمايدۇ. ",
"output": "闭关自守是无法使国家富强的。"
},
```

SFT Data 2 Uy→Zh_Uy
```
{
"instruction": "تۆۋەندىكى ئۇيغۇرچە جۈملىنى دۆلەت تىلىغا تەرجىمە قىلىڭ.",
"input": "بېكىنمىچىلىك قىلىپ دۆلەتنى قۇدرەت تاپقۇزغىلى بولمايدۇ. ",
"output": "闭关自守是无法使国家富强的。"
},
```

SFT Data 3 Zh→Uy_Uy
```
{
"instruction": "تۆۋەندىكى دۆلەت تىلى جۈملىسىنى ئۇيغۇرچىغا تەرجىمە قىلىڭ.",
"input": "闭关自守是无法使国家富强的。 ",
"output": "بېكىنمىچىلىك قىلىپ دۆلەتنى قۇدرەت تاپقۇزغىلى بولمايدۇ."
},
```

SFT Data 4 Zh→Uy_Zh
```
{
"instruction": "请将以下汉语句子翻译成维吾尔语。",
"input": "闭关自守是无法使国家富强的。 ",
"output": "بېكىنمىچىلىك قىلىپ دۆلەتنى قۇدرەت تاپقۇزغىلى بولمايدۇ."
},
```

Figure 5: The example demonstration of the SFT dataset construction, with four sub-figures representing the SFT datasets used in four translation scenarios.

C.1 Uyghur Translation Testing for LLMs

The supplementary results of the Uyghur test for large models are shown in Table 8. According to our supplementary analysis of multiple indicators, it can be seen that, besides the GPT series and Baichuan, which can perform simple Uyghur to Chinese translations, other representative models do not have reasonable Uyghur to Chinese translation capabilities.

C.2 Efficacy of Prompt

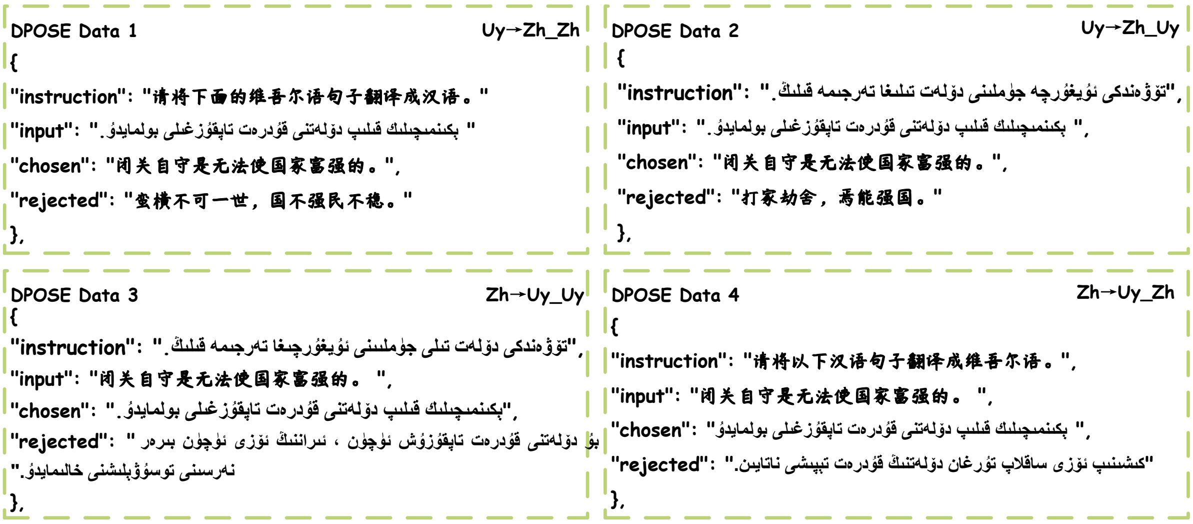
As shown in Table 9, we present a supplementary experiment on the effect of prompts in two scenarios of Uy→Zh, indicating that prompts that are consistent with the target language can bring about greater improvements across multiple metrics.

Model	BLEU5-SBP	BLEU5	BLEU6	NIST6	NIST7	GTM	mWER	mPER	ICT	METEOR	TER
LLaMA2-7B (Touvron et al., 2023b)	0.0005	0.0006	0	0.4517	0.4517	0.0809	1.0463	1.0085	0.01	0.0265	1.5612
ChatGLM3-6B (Du et al., 2022)	0.0009	0.001	0	0.5301	0.5301	0.0946	1.0018	0.9662	0.0081	0.0368	1.6787
Chinese LLaMA2-7B (Cui et al., 2024)	0.0012	0.0014	0	0.6049	0.6049	0.0954	1.07	1.0252	0.0128	0.0349	1.4801
Qwen-7B (Bai et al., 2023)	0.0028	0.0029	0.0014	0.746	0.746	0.1336	0.9588	0.9132	0.0126	0.0589	1.8182
Qwen1.5-14B (Team, 2024)	0.0301	0.0305	0.0211	2.1876	2.1877	0.2779	0.8874	0.77	0.0313	0.1577	1.1459
Baichuan2-13B (Yang et al., 2023)	0.0724	0.0757	0.057	3.3945	3.3948	0.376	0.8373	0.7059	0.0515	0.2591	1.1302
GPT-3.5 (Ouyang et al., 2022)	0.0763	0.0794	0.0589	3.748	3.7482	0.3903	0.8326	0.6675	0.0755	0.2528	0.8847
GPT-4 (OpenAI et al., 2024)	**0.1483**	**0.1527**	**0.1217**	**5.4048**	**5.4059**	**0.5102**	**0.7113**	**0.5449**	**0.1168**	**0.3741**	**0.7301**

Table 8: Supplement on the Uy→Zh LLMs translation results.

Model	Uy→Zh_Uy										
	BLEU5-SBP	BLEU5	BLEU6	NIST6	NIST7	GTM	mWER	mPER	ICT	METEOR	TER
Qwen-7B (Bai et al., 2023)	0.0000	0.0000	0.0000	0.0188	0.0188	0.0090	1.0201	1.0173	0.0605	0.0009	2.9663
LLaMA2-7B (Touvron et al., 2023b)	0.0000	0.0000	0.0000	0.0278	0.0278	0.0092	1.0846	1.0814	0.0210	0.0026	2.1072
Chinese LLaMA2-7B (Cui et al., 2024)	0.0000	0.0000	0.0000	0.0305	0.0305	0.0181	1.0679	1.0621	0.0219	0.0032	2.0492
ChatGLM3-6B (Du et al., 2022)	0.0000	0.0000	0.0000	0.0913	0.0913	0.0181	1.0991	1.0945	0.0278	0.0015	1.6434
Qwen1.5-14B (Team, 2024)	0.0015	0.0018	0.0011	0.2223	0.2223	0.0389	1.0570	1.0436	0.0251	0.0073	2.7451
Baichuan2-13B (Yang et al., 2023)	0.0301	0.0440	0.0328	2.0098	2.0102	0.1975	1.1596	1.0753	0.0239	0.0959	2.0171
GPT-3.5 (Ouyang et al., 2022)	0.0589	0.0612	0.0407	3.628	3.6283	0.3814	0.8609	0.6983	0.0723	0.2296	0.8981
GPT-4 (OpenAI et al., 2024)	0.1159	0.1243	0.0973	5.1298	5.1305	0.4809	0.8263	0.6349	0.0942	0.3171	0.7623
Uyghur LLaMA PT	0.1627	0.1799	0.1554	4.7234	4.7251	0.4158	0.7950	0.6886	0.1722	0.3333	0.8670
Uyghur LLaMA SFT	0.2334	0.2574	0.2210	6.8389	6.8417	0.5784	0.6747	0.5275	0.2353	0.4724	0.6739
Uyghur LLaMA DPOSE	**0.2691**	**0.2896**	**0.2495**	**7.7731**	**7.7760**	**0.6200**	**0.6143**	**0.4529**	**0.2403**	**0.4949**	**0.5513**
Model	**Uy→Zh_Zh**										
	BLEU5-SBP	**BLEU5**	**BLEU6**	**NIST6**	**NIST7**	**GTM**	**mWER**	**mPER**	**ICT**	**METEOR**	**TER**
LLaMA2-7B (Touvron et al., 2023b)	0.0005	0.0006	0.0000	0.4517	0.4517	0.0809	1.0463	1.0085	0.01	0.0265	1.5612
ChatGLM3-6B (Du et al., 2022)	0.0009	0.001	0.0000	0.5301	0.5301	0.0946	1.0018	0.9662	0.0081	0.0368	1.6787
Chinese LLaMA2-7B (Cui et al., 2024)	0.0012	0.0014	0.0000	0.6049	0.6049	0.0954	1.07	1.0252	0.0128	0.0349	1.4801
Qwen-7B (Bai et al., 2023)	0.0028	0.0029	0.0014	0.746	0.746	0.1336	0.9588	0.9132	0.0126	0.0589	1.8182
Qwen1.5-14B (Team, 2024)	0.0301	0.0305	0.0211	2.1876	2.1877	0.2779	0.8874	0.77	0.0313	0.1577	1.1459
Baichuan2-13B (Yang et al., 2023)	0.0724	0.0757	0.057	3.3945	3.3948	0.376	0.8373	0.7059	0.0515	0.2591	1.1302
GPT-3.5 (Ouyang et al., 2022)	0.0763	0.0794	0.0589	3.748	3.7482	0.3903	0.8326	0.6675	0.0755	0.2528	0.8847
GPT-4 (OpenAI et al., 2024)	0.1483	0.1527	0.1217	5.4048	5.4059	0.5102	0.7113	0.5449	0.1168	0.3741	0.7301
Uyghur LLaMA PT	0.1890	0.2089	0.1803	5.4703	5.4721	0.4668	0.7561	0.6347	0.1908	0.3745	0.7797
Uyghur LLaMA SFT	0.2515	0.2689	0.2286	7.6230	7.6250	0.6104	0.6205	0.4611	0.2408	0.4794	0.5600
Uyghur LLaMA DPOSE	**0.2748**	**0.2943**	**2540**	**7.8384**	**7.8411**	**0.6250**	**0.6051**	**0.4437**	**0.2456**	**0.5011**	**0.5432**

Table 9: Supplement on the test on the impact of the language of the prompt on the translation between Chinese and Uyghur.



```
DPOSE Data 1                                    Uy→Zh_Zh
{
"instruction": "请将下面的维吾尔语句子翻译成汉语。"
"input": "بىكىنمىچىلىك قىلىپ دۆلەتنى قۇدرەت تاپقۇزغىلى بولمايدۇ. "
"chosen": "闭关自守是无法使国家富强的。",
"rejected": "蛮横不可一世，国不强民不稳。"
},
```

```
DPOSE Data 2                                    Uy→Zh_Uy
{
"instruction": "تۆۋەندىكى ئۇيغۇرچە جۈملىنى دۆلەت تىلىغا تەرجىمە قىلىڭ.",
"input": "بىكىنمىچىلىك قىلىپ دۆلەتنى قۇدرەت تاپقۇزغىلى بولمايدۇ. ",
"chosen": "闭关自守是无法使国家富强的。",
"rejected": "打家劫舍，焉能强国。"
},
```

```
DPOSE Data 3                                    Zh→Uy_Uy
{
"instruction": "تۆۋەندىكى دۆلەت تىلى جۈملىسىنى ئۇيغۇرچىغا تەرجىمە قىلىڭ.",
"input": "闭关自守是无法使国家富强的。 ",
"chosen": "بىكىنمىچىلىك قىلىپ دۆلەتنى قۇدرەت تاپقۇزغىلى بولمايدۇ.",
"rejected": " بۇ دۆلەتنى قۇدرەت تاپقۇزۇش ئۈچۈن ، ئىراننىڭ ئۆزى ئۈچۈن بىرەر
نەرسىنى توسۇۋېلىشنى خالىمايدۇ."
},
```

```
DPOSE Data 4                                    Zh→Uy_Zh
{
"instruction": "请将以下汉语句子翻译成维吾尔语。",
"input": "闭关自守是无法使国家富强的。 ",
"chosen": "بىكىنمىچىلىك قىلىپ دۆلەتنى قۇدرەت تاپقۇزغىلى بولمايدۇ ",
"rejected": "كىشىنىپ ئۆزى ساقلاپ تۇرغان دۆلەتنىڭ قۇدرەت تېپىشى ناتايىن."
},
```

Figure 6: The example demonstration of the DPOSE dataset construction, with four sub-figures representing the DPOSE datasets used in four translation scenarios.