Relay Decoding: Concatenating Large Language Models for Machine Translation

Anonymous ACL submission

Abstract

Leveraging large language models for machine translation has demonstrated promising results. However, it does require the large language models to possess the capability of handling both the source and target languages in machine translation. When it is challenging to find large models that support the desired languages, resorting to continuous learning methods becomes a costly endeavor. To mitigate these expenses, we propose an innovative approach called **RD** (**R**elay **D**ecoding), which entails concatenating two distinct large models that individually support the source and target languages. By incorporating a simple mapping layer to facilitate the connection between these two models and utilizing a limited amount of parallel data for training, we successfully achieve superior results in the machine translation task. Experimental results conducted on the Multi30k and WikiMatrix datasets validate the effectiveness of our proposed method.[1]

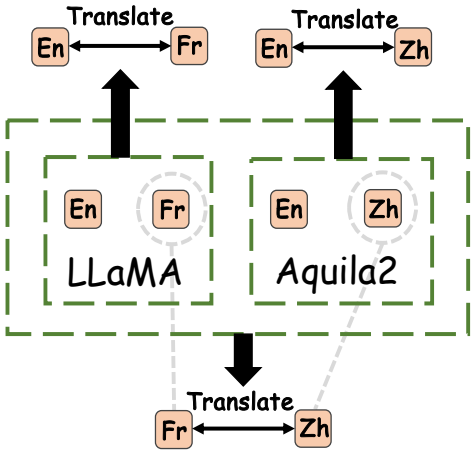

Figure 1: LLaMA's supported languages include English and French while Aquila mainly support English and Chinese. Both Aquila2 and LLaMA are not proficient in handling the Chinese to French translation task individually. In such cases, we can concatenate the two models to accomplish the translation task.

1 Introduction

The remarkable capabilities of large language models (LLMs) with billions of parameters have been demonstrated across various tasks. Several studies have leveraged these LLMs to accomplish and enhance machine translation tasks (Zhang, 2023; Li, 2023; Garcia et al., 2023; Jiao et al., 2023; Lyu et al., 2023; Huang et al., 2024). Through techniques such as In-Context Learning(ICL), Chain-of-Thought(COT) and Instructions Finetuning, these LLMs have been able to achieve translation abilities comparable to state-of-the-art machine translation systems.

However, the use of LLMs for translation is still limited by the languages supported by these models. Frequently, it is challenging to find LLMs that can effectively support both the source language L_a and target languages L_b simultaneously, which poses a significant limitation. In such a scenario, one direct approach is to further train the existing LLM, which primarily supports one language, to incorporate the capabilities of another language(Cui et al., 2023). However, this requires an enormous amount of pretrained data and poses significant challenges due to the large framework of the model. Additionally, it is crucial to ensure that catastrophic forgetting does not occur, preserving the proficiency of the model in its original language.

Are there any strategies to mitigate the costly nature of continuous learning? We have observed that, in practice, it is relatively straightforward to acquire LLMs that excel exclusively in either the source or target languages. As shown in Figure 1, by concatenating these specialized language models, it becomes conceivable to achieve translation without incurring the exorbitant expenses associated with continuous learning. In exceptional scenar-

[1]The dataset and associated codes will be publicly available.

ios, when confronted with languages that lack pre-existing LLMs, a viable approach involves training a monolingual large model from scratch for the specific language. Subsequently, employing a concatenation technique enables us to accomplish machine translation, while also circumventing the issue of catastrophic forgetting in continuous learning.

Drawing on these insights, we propose a simple yet effective method **RD** (**R**elay **D**ecoding) for large model concatenation to achieve machine translation, where each large model specifically supports the source and target languages of the translation task. RD utilizes a simple mapping layer to connect two LLMs, leveraging a small portion of parallel corpora to train this mapping layer. In our experiments conducted on datasets such as Multi30k and WikiMatrix, utilizing the LLaMA and Aquila2 models, we find that our approach surpasses the method of fine-tuning with a single large model. Furthermore, we observed significant improvements of over 3 BLEU points in certain language pairs.

2 Approach

2.1 Task Description

For a translation task from Language L_a to Language L_b, when it is not possible to find a single large model that performs well for both languages simultaneously, we focus on finding a separate large model for each language that excels in that specific language. Let M_a denote a large language model that excels in language L_a, and M_b denote another large language model that excels in language L_b. RD aims to concatenate M_a and M_b to achieve the translation task from L_a to L_b.

2.2 Concatenate Method

As illustrated in Figure 2, for a given sentence $X = \{x^1, x^2, ..., x^K\}$ in language L_a containing K tokens, we utilize M_a to decode and generate its corresponding representation, denoted as $H \in \mathcal{R}^{K*D_h}$. D_h is the hidden states size of M_a. Subsequently, we utilize a mapping function $W_p \in \mathcal{R}^{D_h*D_e}$ to project the obtained hidden representation H into the input space of M_b. D_e is the embedding layer size of M_b. For the sake of simplicity and efficiency, we employ a linear layer as the mapping layer[2], similar to the connection

[2]We also attempt alternative methods of connecting the structures, which are documented in Appendix A.

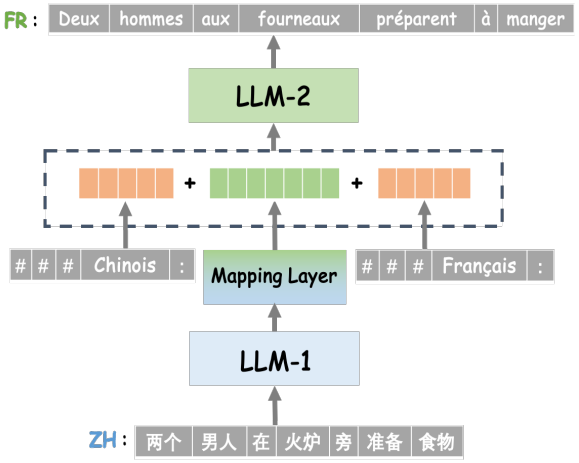

Figure 2: Using Chinese-French translation as a case in point for the process of Relay Decoding.

methods used in many multi-modal large models (Koh et al., 2023; Zhang et al., 2023c,a).

Next, we introduce a prompt to facilitate better generation by M_b. When the source language is Chinese and the target is English, the prompt would be as follows:

$$\#\#\#[Chinese] : X \quad \#\#\#[English] :$$

The pattern $\#\#\#[\star]$ is employed to denote the name of the specific language. In our case, we use the target language to replace these patterns.

After tokenizing the prompt and passing it through the embedding layer of M_b, we obtain the prompt input representation E_p. Finally, we concatenate the mapped representations H with the prompt representations E_p and feed them into M_b for further decoding and generation.

2.3 Training Strategy

We formulate translation task as generating target text tokens conditioned on a source text tokens and prompt prefix. The log likelihood of target sentence Y (tokenized as $\{y_1, y_2, .., y_T\}$) conditioned on its source sentence X is:

$$l(X, Y) = \sum_{t=1}^{T} log P_\theta(y_t | H, E_p, y_1, y_2, ..., y_{t-1})$$

The loss L is then the negative log likelihood of all samples in a batch of N bilingual parallel pairs:

$$L = -\frac{1}{N} \sum_{i=1}^{N} l(X_i, Y_i)$$

Method	Zh-Fr		Zh-De		Zh-Cs	
	BLEU	chrF	BLEU	chrF	BLEU	chrF
Bilingual	20.70	47.5	10.82	38.3	7.52	27.7
Aquila2	19.65	49.0	10.32	43.0	8.75	36.1
LLaMA	25.76	53.3	15.00	49.3	10.08	38.6
RD (Aquila2+LLaMA)	**27.36**	**55.1**	**17.87**	**49.8**	**13.44**	**39.1**

Table 1: The result of RD Method for Zh-Fr, Zh-De, Zh-Cs translation tasks on Multi30k dataset.

When utilizing only one LLM for translation, finetuning has been shown to yield optimal results. Therefore, in our approach, we also experiment with incorporating finetuning, which involves simultaneously adjusting the parameters of the large model during the training process. To prevent the occurrence of catastrophic forgetting, we introduced LORA(Hu et al., 2021) as a mechanism to mitigate this challenge.

3 Experiment

In this section, we provide a description of the datasets, experimental setup employed in our study and an in-depth analysis of the results obtained.

3.1 Experimental Details

Large Language Models The LLMs utilized in our experiments include the Aquila2-7B model [3] and the LLaMA-7B model (Touvron et al., 2023). In our experiments, we primarily focus on translation from Chinese to French, German, and Czech. The Aquila2 model primarily focuses on English and Chinese proficiency and performs remarkably well in tasks involving these languages. On the other hand, the LLaMA model has been pretrained on datasets encompassing twenty languages, such as English, French, German, and Czech, but its Chinese proficiency is relatively lower.

Datasets The datasets used in our experiments are Multi30k dataset (Elliott et al., 2016, 2017; Barrault et al., 2018) and WikiMatrix dataset (Schwenk et al., 2019). Multi30k dataset contains images and their captions in four languages: English(En), French(Fr), Germany(De), and Czech(Cs). For Chinese translation task, we have annotated a Chinese version of the Multi30k dataset[4]. Initially, we employ ChatGPT[5] to translate the English content of the dataset into Chinese. Subsequently, we conduct manual revisions to address any inaccuracies or lack of fluency in the translation. As for test set, we use Flickr2017 for Zh-Fr and Zh-De and Flickr2018 for Zh-Cs. Regarding WikiMatrix, we specifically choose the Simplified Chinese portion of the dataset. From this subset, we select the top 1000 highest-scoring pairs as our test set, while the remaining pairs are used for training.

Baselines We compared our method with the following approaches: (1) Transformer-based bilingual translation model. (2) Results of instruction fine-tuning large models, including LLaMA and Aquila2. That's an important point to note that while LLaMA may have lower proficiency in Chinese, it still has some capability in handling and generating Chinese due to the presence of a portion of Chinese data in its training set. Similarly, Aquila2 model's training corpus may also include a small amount of French, German, and Czech data. As a result, fine-tuning directly on these models can still achieve some level of performance in translation tasks for the respective languages.

3.2 Main Results

The experimental results on Multi30k dataset for Zh-Fr, Zh-De, and Zh-Cs are presented in Table 1. From the table, we observe that our RD method achieves the best results. When comparing the last three rows with the first row, which represents the bilingual transformer approach, we find that utilizing large models with the same parallel corpus outperforms training from scratch. This indicates that the language alignment capability of the large models is indeed utilized during training, even though they were pretrained only on monolingual data. The results of fine-tuning large models specialized in one language (rows 2 and 3) show that these models still have some limitations in completing translation tasks. Additionally, we have also discovered that LLMs specialized in the target

[3] https://github.com/FlagAI-Open/Aquila2.
[4] The Chinese version of multi30k will be available.
[5] https://chat.openai.com/.

Aquila2 (ZH)	LLaMA (FR)	Zh-Fr
Not Finetune	Not Finetune	25.94
Not Finetune	Finetune	**27.36**
Finetune	Not Finetune	23.37
Finetune	Finetune	26.88

Table 2: The BLEU scores of different finetune settings on Multi30k dataset.

#Dataset	2W	3W	4W
RD	11.79	12.98	13.64
#Dataset	**5W**	**6W**	**7W**
RD	14.26	15.44	15.52

Table 3: The BLEU scores associated with varying WikiMatrix dataset sizes.

language tend to exhibit superior performance in translation tasks. Our concatenation method also surpasses the performance of fine-tuning with a single large model, demonstrating the need for pre-training large models on both the source and target languages to achieve better translation performance and this further validates the effectiveness of our proposed concatenation method.

3.3 Analysis

Is it necessary to finetune the LLMs during training? Our approach involves training a mapping layer to connect two large models, but during training, we also need to consider whether to adjust the parameters of the large models. As shown in Table 2, we test the translation performance of Zh-Fr under different finetuning conditions on Multi30K datasets and find that simultaneously finetuning the parameters of the second large model (i.e., the one specialized in the target language) yield better results. On the other hand, fine-tuning the parameters of the first large model has a less significant impact. For finetuning, we utilized the efficient fine-tuning method known as LORA due to its higher efficiency.

How much data is required to complete the training of the mapping layer? As presented in Table 3, we conducted Zh-Fr translation experiments using training sets of different sizes on WikiMatrix datasets. The findings reveal that on the WikiMatrix dataset, training with approximately 60,000 data points is adequate for training the mapping layer. This requirement is considerably smaller compared to the dataset size typically needed by traditional bilingual methods. Moreover, our method surpasses these methods in performance.

4 Related Work

LLMs for Machine Translation. With the remarkable advancements of LLMs, researchers have extensively evaluated their translation capabilities using various methodologies. Vilar et al. (2023); Zhang et al. (2023b); Bawden and Yvon (2023) devise different prompts to facilitate translation and also examine the translation performance in various few-shot scenarios. Peng et al. (2023); Huang et al. (2024) utilize Chain-of-thought or difficulty analysis techniques to address translation challenges. In order to achieve better performance, Li et al. (2023); Jiao et al. (2023); Chen et al. (2023); Alves et al. (2023); Xu et al. (2023) have explored the approach of finetuning LLMs using parallel data. All of the aforementioned methods require full support from the LLMs for the languages involved in translation. Our approach, on the other hand, is primarily designed for situations where a single large language model cannot handle all of these languages simultaneously.

Concatenation of LLMs. Bansal et al. (2024) leverages the concatenation of a smaller model and a larger model to augment the capabilities of the larger one, such as enhancing low-resource language comprehension and mathematical computation abilities. In comparison, our concatenation method is specifically tailored for machine translation tasks. Furthermore, unlike our method, they do indeed require both models to be capable of handling vocabulary from both languages involved in the translation task.

5 Conclusion and Future Work

In this paper, we propose an approach that involves concatenating two LLMs, each specialized in the source and target languages, to achieve machine translation. This method circumvents the higher costs associated with continuous learning approaches. In the future, we plan to delve deeper into this concatenation method and investigate how to accomplish the connection solely with monolingual data as the finetuning approach for LLMs does not necessitate the use of bilingual data.

Limitations

In our concatenation approach, we require a certain amount of parallel data to train the parameters of the concatenation module. Acquiring parallel data can be costly, so in the future, we plan to explore methods that rely on monolingual data and back-translation to train the parameters of the concatenation module.

References

Duarte Alves, Nuno Guerreiro, João Alves, José Pombal, Ricardo Rei, José de Souza, Pierre Colombo, and Andre Martins. 2023. Steering large language models for machine translation with finetuning and in-context learning. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 11127–11148, Singapore. Association for Computational Linguistics.

Rachit Bansal, Bidisha Samanta, Siddharth Dalmia, Nitish Gupta, Sriram Ganapathy, Abhishek Bapna, Prateek Jain, and Partha Talukdar. 2024. LLM augmented LLMs: Expanding capabilities through composition. In *The Twelfth International Conference on Learning Representations*.

Loïc Barrault, Fethi Bougares, Lucia Specia, Chiraag Lala, Desmond Elliott, and Stella Frank. 2018. Findings of the third shared task on multimodal machine translation. In *Proceedings of the Third Conference on Machine Translation: Shared Task Papers*, pages 304–323.

Rachel Bawden and François Yvon. 2023. Investigating the translation performance of a large multilingual language model: the case of BLOOM. In *Proceedings of the 24th Annual Conference of the European Association for Machine Translation*, pages 157–170, Tampere, Finland. European Association for Machine Translation.

Yijie Chen, Yijin Liu, Fandong Meng, Yufeng Chen, Jinan Xu, and Jie Zhou. 2023. Improving translation faithfulness of large language models via augmenting instructions. *arXiv preprint arXiv:2308.12674*.

Yiming Cui, Ziqing Yang, and Xin Yao. 2023. Efficient and effective text encoding for chinese llama and alpaca. *arXiv preprint arXiv:2304.08177*.

Desmond Elliott, Stella Frank, Loïc Barrault, Fethi Bougares, and Lucia Specia. 2017. Findings of the second shared task on multimodal machine translation and multilingual image description. In *Proceedings of the Second Conference on Machine Translation*, pages 215–233, Copenhagen, Denmark. Association for Computational Linguistics.

Desmond Elliott, Stella Frank, Khalil Sima'an, and Lucia Specia. 2016. Multi30K: Multilingual English-German image descriptions. In *Proceedings of the 5th Workshop on Vision and Language*, pages 70–74, Berlin, Germany. Association for Computational Linguistics.

Xavier Garcia, Yamini Bansal, Colin Cherry, George Foster, Maxim Krikun, Melvin Johnson, and Orhan Firat. 2023. The unreasonable effectiveness of few-shot learning for machine translation. In *International Conference on Machine Learning*, pages 10867–10878. PMLR.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*.

Yichong Huang, Xiaocheng Feng, Baohang Li, Chengpeng Fu, Wenshuai Huo, Ting Liu, and Bing Qin. 2024. Aligning translation-specific understanding to general understanding in large language models. *arXiv preprint arXiv:2401.05072*.

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.

Jing Yu Koh, Ruslan Salakhutdinov, and Daniel Fried. 2023. Grounding language models to images for multimodal inputs and outputs. In *International Conference on Machine Learning*, pages 17283–17300. PMLR.

Jiahuan Li, Hao Zhou, Shujian Huang, Shanbo Chen, and Jiajun Chen. 2023. Eliciting the translation ability of large language models via multilingual finetuning with translation instructions. *arXiv preprint arXiv:2305.15083*.

Yinheng Li. 2023. A practical survey on zero-shot prompt design for in-context learning. In *Proceedings of the 14th International Conference on Recent Advances in Natural Language Processing*, pages 641–647, Varna, Bulgaria. INCOMA Ltd., Shoumen, Bulgaria.

Chenyang Lyu, Jitao Xu, and Longyue Wang. 2023. New trends in machine translation using large language models: Case examples with chatgpt. *arXiv preprint arXiv:2305.01181*.

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.

Keqin Peng, Liang Ding, Qihuang Zhong, Li Shen, Xuebo Liu, Min Zhang, Yuanxin Ouyang, and Dacheng Tao. 2023. Towards making the most of

ChatGPT for machine translation. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 5622–5633, Singapore. Association for Computational Linguistics.

Holger Schwenk, Vishrav Chaudhary, Shuo Sun, Hongyu Gong, and Francisco Guzmán. 2019. Wikimatrix: Mining 135m parallel sentences in 1620 language pairs from wikipedia. *arXiv preprint arXiv:1907.05791*.

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

David Vilar, Markus Freitag, Colin Cherry, Jiaming Luo, Viresh Ratnakar, and George Foster. 2023. Prompting PaLM for translation: Assessing strategies and performance. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15406–15427, Toronto, Canada. Association for Computational Linguistics.

Haoran Xu, Young Jin Kim, Amr Sharaf, and Hany Hassan Awadalla. 2023. A paradigm shift in machine translation: Boosting translation performance of large language models. *arXiv preprint arXiv:2309.11674*.

Ao Zhang, Hao Fei, Yuan Yao, Wei Ji, Li Li, Zhiyuan Liu, and Tat-Seng Chua. 2023a. Transfer visual prompt generator across llms. *arXiv preprint arXiv:2305.01278*.

Biao Zhang, Barry Haddow, and Alexandra Birch. 2023b. Prompting large language model for machine translation: A case study. *arXiv preprint arXiv:2301.07069*.

Jia Zhang. 2023. Exploring undergraduate translation students' perceptions towards machine translation: A qualitative questionnaire survey. In *Proceedings of Machine Translation Summit XIX, Vol. 2: Users Track*, pages 1–10, Macau SAR, China. Asia-Pacific Association for Machine Translation.

Yanzhe Zhang, Ruiyi Zhang, Jiuxiang Gu, Yufan Zhou, Nedim Lipka, Diyi Yang, and Tong Sun. 2023c. Enhanced visual instruction tuning for text-rich image understanding. In *NeurIPS 2023 Workshop on Instruction Tuning and Instruction Following*.

A Mapping Layers

We explored three different approaches for achieving the mapping as shown in Figure3:

(1) Linear: Directly employing a linear layer, as previously mentioned, denoted by FC.

Mapping Method	FC	CA	CA-Q
Zh-Fr	27.36	11.80	17.92

Table 4: The BLEU score of different mapping method on Multi30k dataset.

(2) Cross-attention: Employing a cross-attention structure with the source language input X, passed through the embedding layer of M_b, serving as the query, denoted by CA. (3) Cross-attention with query embedding: Utilizing a cross-attention structure with randomly initialized query embeddings, denoted by $CA-Q$.

We also conducted Zh-Fr translation experiments on the Multi30k dataset, and the experimental results are presented in Table 4.

We observe that the Linear mapping method achieved the best results on the Multi30k dataset, while the cross-attention series method yield lower results, even lower than the baseline methods. This could be attributed to the larger number of parameters introduced by these methods, which may not be effectively learned due to the relatively small scale of the Multi30k dataset.

B Experiments System Settings and Evaluation Metric

We use Adam optimizer and 2000 warm-up updates. The learning rate is 1e-5. For evaluation, we use 4-gram BLEU (Papineni et al., 2002) and chrF scores by multi-bleu.pl in Moses[6]. We train all models on NVIDIA 80GB A100 GPUs.

[6]https://github.com/moses-smt/mosesdecoder

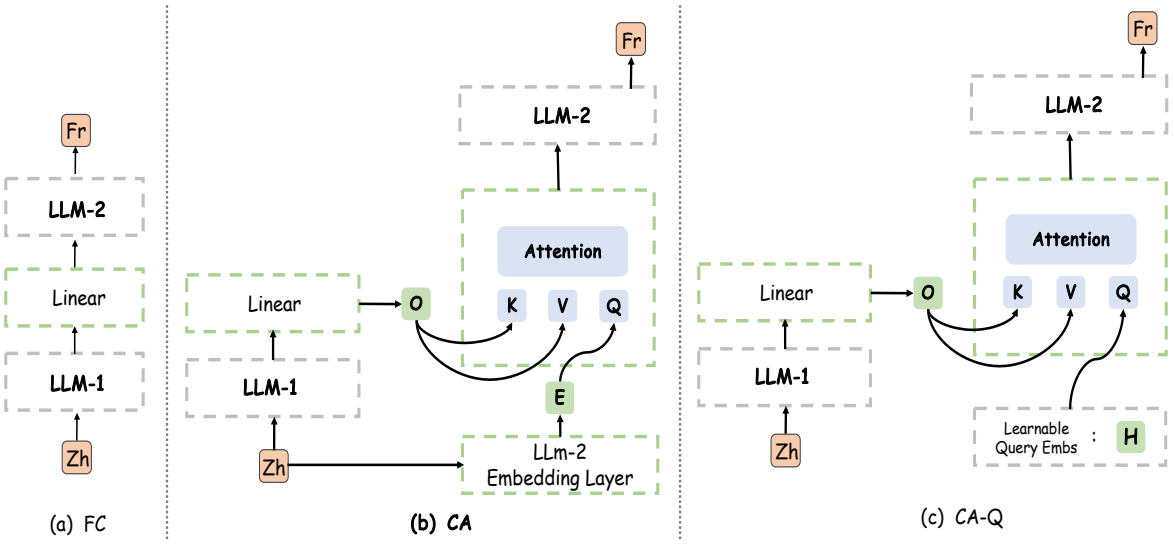

Figure 3: Different mapping layers.