Enhancing Machine Translation in Low-Resource Languages: A Comparative Review of Prompt-Based and Fine-Tuning Methods

Author: [Author name redacted by Managing Editor]

Abstract:

Large language models (LLMs) such as LLama, GPT, T5, and Alpaca have demonstrated capabilities in multilingual tasks with natural language processing (NLP). However, low-resource languages often lack data regarding tokens and linguistic representations, which are critical for effective training of the models. This limits the models' performance, particularly in translation, compared to high-resource language models, which benefit from large labeled datasets. There is clearly a need to fix the imbalance of effectiveness of LLMs in low-resource languages. While many methods exist to address this problem, it is unclear which method is best. In this review, we evaluate methods recently developed that attempt to enhance the translation performance of LLMs in low-resource languages. Specifically, we focus on prompt-based and fine-tuning methods - methods that have been shown to significantly improve the performance of LLMs in low-resource languages. Among the methods reviewed, Parameter-Efficient Fine-Tuning (PEFT)—including Adapters and Prefix Tuning—demonstrated the best balance between performance and cost, making them the most practical for low-resource languages. However, traditional full fine-tuning methods, despite higher computational requirements, achieve greater performance improvements, emphasizing the importance of balancing performance gains with resource constraints. These findings highlight the importance of balancing architectural enhancement methods with culturally and quantity-limited data.

1. Intro

Prior works of adapting LLMs to specific domains used full fine-tuning, which can be costly and lacks effectiveness, specifically for low-resource languages. Methods to overcome these issues/disadvantages have been proposed. However, with the current state, there are too many options to choose from.

The fast-developing technology of large language models (LLMs) like Llama, GPT, T5, and Alpaca has transformed how humans interact with machines through content creation and human-like dialogue (Qin et al., 2024). For example, generative LLMs can generate content from any user input, such as text, images, sound, etc. These models operate using transformer-based architectures and are trained on extensive text corpora to identify linguistic patterns and accurately predict subsequent tokens. This enables LLMs to generate natural and coherent texts within a very short time.

Benefits of LLMs are not equally distributed across the world. A major contributor to this is the disparity of performance across different languages. The majority of LLMs are trained on high-resource languages, primarily English, where large amounts of text data are easily accessible. Therefore, LLMs respond significantly better to English commands and prompts than to languages that have less publicly accessible corpora- 'low-resource' languages (Ming et al., 2024; *(PDF) Multilingual Large Language Models and Curse of Multilinguality*, 2025). Approximately, there are 3 billion people worldwide who speak low-resource languages, and the differences in performance and usability cause inequality in linguistic representation and opportunity. For example, LLMs trained only in English cannot serve as a highly useful tool when it comes to assisting local governments, indigenous communities. For example, in one evaluation of translation tasks, English prompts achieved an average BLEU score of 92.5%, while Hindi prompts scored only 7.5% ("(PDF) Multilingual Prompting in LLMs," 2024). BLEU is a standard metric that evaluates how closely machine translations match human translations. Higher scores mean better translation quality(Papineni et al., 2002).

Researchers have tackled this problem by using the multilingual pretraining method (such as macroLLM), which involves training LLMs on texts from a mixture of languages to facilitate knowledge transfer across different languages (Ming et al., 2024). This approach enhances the models' ability to understand and generate low-resource language texts by leveraging knowledge from high-resource languages. However, this method caused a regression of the model's ability to respond to high-resource languages ((PDF) *Multilingual Large Language Models and Curse of Multilinguality*, 2025). Additionally, since the multilingual pretraining method requires large amounts of data for training, the imbalance in data

availability between high-resource and low-resource languages leads to uneven performance.

Recently novel approaches have been developed to address this problem. Fine-tuning methods are methods that involve a direct adjustment of parameters (weights and biases) of the model, which means that the model itself is modified for domain-specific tasks. They can be categorized into two different types, as seen in Figure 1: either a full fine-tuning method, which involves updating all of the parameters, or parameter-efficient fine-tuning methods (PEFT), which only adjust some of the parameters. Examples include Adapters, Prefix Tuning, and Self-Distillation from Resource-Rich Languages (SDRRL)(Ding et al., 2023; Houlsby et al., 2019; Li & Liang, 2021; Zhang et al., 2024).

On the other hand, prompt-based methods are methods that do not require direct modification of the parameters of the models. Instead, this methods add extra tokens to the prompt that guide the model to the correct answer by reducing the possibility of different interpretations, increasing precision. This includes Dictionary-based phrase level Prompting Machine Training (DiPMT), Retrieved Phrase level Prompts (RePP), prompt tuning(Ghazvininejad et al., 2023; Sun et al., 2022; Lester et al., 2021).

In this work, we qualitatively evaluate recent prompt-based and fine-tuning methods for low-resource machine translation (MT), identifying optimal strategies based on performance, practicality, and cost-effectiveness.

2. Methodology

A literature review was conducted to identify relevant research on methods enhancing the performance of large language models (LLMs) for low-resource languages. Databases including Google Scholar and ACL anthology were comprehensively searched using keywords such as “low-resource languages”, “Multilingual large language model”, “prompt-based methods”, “Word Type coverage”and “parameter efficient fine tuning.”

The papers reviewed were selected based on clearly defined criteria. Each paper needed to clearly focus on adapting large language models, empirically evaluate

either prompt-based methods or fine-tuning methods, include detailed evaluations regarding performance, computational efficiency, and costs, be peer-reviewed, and written in English in a text format. Our goals were to identify recent methods specifically aimed at enhancing MT performance in low-resource languages, critically evaluate these methods using criteria such as computational practicality, implementation cost, and effectiveness, and to recommend the most suitable methods for practical implementation, considering restricted data constraints and linguistic variations in low-resource languages. The cost was calculated based on the total number of parameters in GPT4.1 and how much was modified by each method.

In this review prompt-based methods were defined as an approach taken with zero modification of the parameters of the original model, focusing on enhancing the prompts to feed into the model to yield better performance. Conversely, fine-tuning methods refer to approaches that directly modify the parameters of the original model, thereby refining its transformer architecture or neural network structure to produce more effective outputs.

3. Results Section

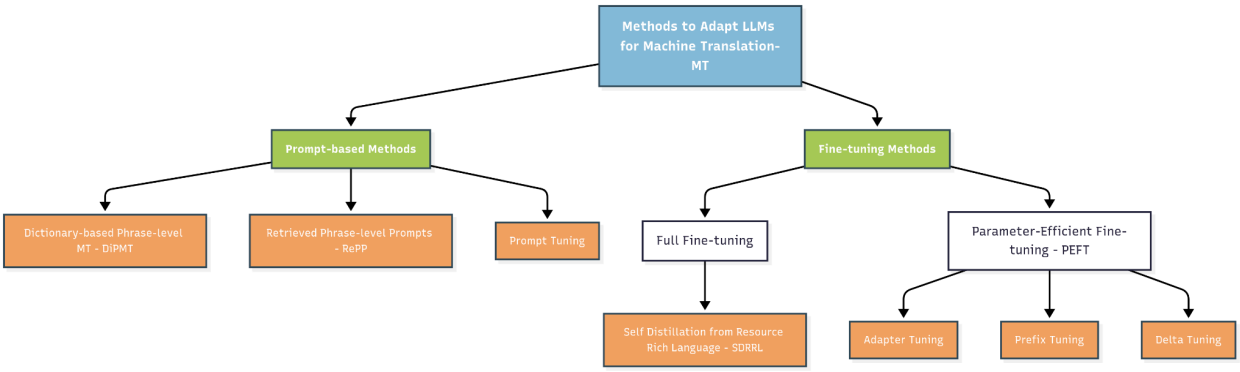

Figure 1. Hierarchy diagram of methods reviewed

Figure 1. Represents the methods we have identified and how it is categorized. The methods is categorized in two large parts Promp-based Methods and Fine-tuning Methods.

3.1 Prompt-based methods:

Prompt-based methods are a way to enhance LLMs' performance by editing the original prompt. Specifically, in prompt-based methods, the original input is modified to fit a template with a specified format designed to clarify the original prompts to the LLM. Each separate method uses distinct templates with unfilled slots and placeholders. These unfilled slots are filled by the original model probabilistically to obtain the final prompt, which is then used to derive the final output (Liu et al., 2021). Unlike traditional fine-tuning, Prompt-based methods do not require additional training, so they show improvements much faster and at a lower cost (Shin et al., 2025).

Initial language models such as N-grams and Recurrent Neural Networks (RNN) laid the foundation for prompt-based methods. Large-scale models such as GPT-3 enhanced the ability of prompt engineering to greater adaptability.

In this section, we compare the following methods proposed by other researchers that were developed recently: Dictionary-based Phrase Level Prompting (DiPMT), Retrieval Phrase Level Prompting (RePP), and Prompt Tuning (Ghazvininejad et al., 2023; Lester et al., 2021; Sun et al., 2022).

3.1.1 DiPMT (Ghazvininejad et al., 2023)

Dictionary-based Phrase-level Machine Translation (DiPMT) was developed in 2023 by Meta AI. DiPMT is a domain specific method specialized at the task of Machine Translation. The core idea behind DiPMT is to enhance prompts using a bilingual dictionary to give hints for specific words, especially rare words, to narrow down the possibility of the interpretation for the LLMs.
In their study, Ghazvininejad et al. evaluated two LLMs—an English-based model (OPT) and a multilingual model (Bloom)—comparing their performance against each other as well as a baseline model. Multilingual models are language models trained on datasets from multiple languages, enabling them to process and generate text across diverse linguistic systems.
In DiPMT, strings describing the meaning of the keywords (e.g. the word "pembuatan" means "creation") are appended to the original prompt. The addition

of these strings expands the model's capability to accurately translate the original prompt. As a result of incorporating this method, the model showed an improvement in the BLEU compared to the baseline LLM output (which omits the dictionary hints from the prompt) with a mean increase average of +4.8, and a standard deviation of ±1.2 across 17 languages of the 20 tested. An example prompt in a DiPMT:

> Translate the following sentence to English: Ia melakukan pembuatan bel pintu dengan teknologi WiFi, katanya.
> In this context, the word "pembuatan" means "creation"; the word "bel" means "buzzer", "bell"; the word "pintu" means "door", "doors".
> The full translation to English is:

Figure 2. Example of a zero-shot prompt using DiPMT. Green words, which acts as a hints, are added to the original blue & red instruction.

This method highly depends on the type coverage of the dictionary for the particular language that is being translated. The Word Type Coverage of a dictionary represents the proportion of distinct words in the language existing in the dictionary. A high Word Type Coverage means the dictionary is robust and covers diverse vocabulary. On the other hand, Word Token Coverage considers the percentage of dictionaries that can be translated in real-world situations. Thus, small dictionaries can have high Token Coverage if they contain frequent common words in the language. This relationship underscores the importance of lexical coverage: when dictionaries include a wide variety of word types, models can better clarify meanings during translation, especially in underrepresented languages.

The study shows that DiPMT only performs extensively better when the Type Coverage is above a certain level (5 - 20%). Typical lexical coverage for low-resource languages is around 20 - 50%, which suggests that DiPMT is practically useful in real-life situations.

The performance of the method is highly dependent on the hint that is provided. The prompts produced by the normal DiPMT method provide all available dictionary translations of words in the original prompt, creating multiple competing options that can be challenging for LLMs to determine the most contextually appropriate choice. The results from the paper show that the output

from this method can sometimes be inaccurate since the model could potentially choose the wrong definition. Although this method can be highly accurate when "gold hints" –which are the correct translations of some key words in the original prompts– are provided, this may not be possible to attain for every case of translations. Thus, a method that would allow gold hints to be derived from dictionary hints will increase the effectiveness of DiPMT significantly; else, the performance may be limited.

Importantly, even if the dictionary definitions are inaccurate, the model's performance does not fall below the baseline—it simply proceeds without relying on the dictionary hints.

3.1.2 RePP (Sun et al., 2022)

Retrieved Phrase-level Prompts (RePP) enhances LLM's outputs by retrieving relevant phrase-level data from a bilingual dictionary database and combining this with the input before processing in the LLM. Additional precision from the database clarifies meanings for the LLM to process, and using phrase-level data limits the possibility of interpretations compared to sentence-level data. There are three stages to this method. The first stage is splitting the original prompt into phrases and extracting the corresponding, input-relevant, bilingual phrases from the pre-built phrase-level database. Then, using the extracted phrases and the original input, a prompt is created. Finally, this prompt is fed to the LLM to produce a translation. Additionally, RePP utilizes a zero-shot method, meaning there are no examples appended to the prompt, to resolve domain adaptation issues.

Sun et al. measured RePP's performance by comparing the translation performance with a baseline model and a full fine-tuned model, specifically for English-German and German-English translation. As a result, the RePP model outperformed the non-trained method by an average of 6.2 BLEU scores, but performed less than the full fine-tuned model.

Compared to DiPMT, RePP does not heavily depend on external dictionary quality, thus making it a more consistent method. This is because the bilingual dictionary

database is a combination of multiple dictionaries. Combining multiple dictionaries will increase Word type coverage and possibly word token coverage as well. However, creating this bilingual dictionary database can be more challenging than DiPMT. A combined bilingual dictionary database demands careful integration and standardization across multiple sources.
The differences in performance primarily arise from the extent and depth of model parameter adjustments; methods like Adapters involve substantial parameter updates, allowing deeper task-specific adaptation, whereas Prompt Tuning, with minimal parameter modifications, provides more limited enhancement.

DiPMT exceeds performance when translating sentences with rare words. This is because RePP does not adapt to the rare words or any word-specific definition. Instead, it focuses on understanding the nuance of sentences, which may be more accurate in translating idioms or long paragraphs (Ghazvininejad et al., 2023; Sun et al., 2022).

Furthermore, RePP only requires one main model, whereas traditional fine-tuning methods require n models for n domains. This is because the input is converted into a prompt, using input-relevant data, which enhances the domain adaptation, ultimately, to derive the output. This suggests their robustness in a low resource situation, likely because not much training can be done with low resource languages due to the lack of corpora, and there might be less investment due to the lack of demand with less number of population using low resource languages in comparison to high resource languages like English.

3.1.3 Prompt Tuning (Lester et al., 2021)

In recent efforts to adapt large language models to specific tasks more efficiently, various parameter-efficient tuning strategies have emerged, one of which is prompt tuning. Prompt tuning freezes the existing parameters in the original model and adds a learnable vector in the input embedding section (bottom layer of the transformer) that guides the input to give the correct output. This learnable vector is prepended on the prompt and adjusted accordingly to the specific task, hence it is called prompt tuning. Once the prompt tuning is complete, and actual

interference occurs, the learned vector acts as an extra guide to understand the language's meanings.

In the presented paper, Lester et al. analyzed prompt tuning's performance using GLUE and SuperGLUE benchmarks, and the model they used was T-5 XXL. As a result, they achieved the same performance as fine-tuning in terms of performance (89.3).

Prompt tuning is one of the cheapest methods available, adjusting only approximately ~0.01% of the entire parameter. Calculating from the average GPT4.1 parameter size, it only costs about $0.75 to prompt tune models.

Moreover, since most of the core language model parameters are frozen, prompt tuning prevents the model from modifying the original understanding of other languages like English. This is critical in Machine Translation as the general understanding of the linguistic characteristics of both languages is critical. Typical fine-tuning could leverage the performance for both languages, but at the same time lower the performance of high-resource languages.

3.2 Fine-Tuning Methods:

Traditional fine-tuning involved retraining the model with domain-specific data to adapt the model to yield better performance at distinct tasks. This method is significantly effective since fine-tuning takes less time and less domain-specific data compared to model architecture modification (Tian et al., 2023).

Full fine-tuning involves the entire parameters of the LLM models to be readjusted. Since full fine-tuning modifies all of the parameters, each fine-tuned model will only be tailored to one specific task. However, as models become upscaled, such as GPT-3 with billions of parameters, fine-tuning becomes heavily costly due to the need for a massive amount of storage for the parameters, and this increases massively with every task-specific model. Furthermore, full fine-tuning requires 2M - 10M tokens to feed to the model (Su et al., 2024), which is difficult to obtain in a low-resource language setting. To tackle this increasing storage issue and minimal data issue, Parameter Efficient Fine Tuning (PEFT) was created. We will evaluate

how these fine-tuning methods can resolve the issues while enhancing the accuracy of the low-resource language translation.

In this section, we compare the following methods proposed by other researchers that were developed recently: Self Distillation from Resource Rich Languages (SDRRL) (Zhang et al., 2024), Parameter Efficient Fine Tuning (PEFT) including Adapters and Prefix tuning (Ding et al., 2023; Houlsby et al., 2019; Li & Liang, 2021).

3.2.1 SDRRL (Zhang et al., 2024)

Self-distillation from Resource Rich languages (SDRRL) is an extension of supervised full fine-tuning that uses cross-lingual transfer from high-resource languages to leverage the multilingual capabilities with low-resource languages as well.

Unlike full fine-tuning, which retrains every model parameter on the low-resource data, SDRRL extracts key patterns learned from high-resource languages and applies them directly to the target language, avoiding a full retraining.

The LLMs answer questions in a high-resource language (such as English) as an exemplar output, hinging on the fact that LLMs perform proficiently in high-resource languages. Then, the generated answers are translated, which are stored in the transfer set –a data set consisting of multilingual pairs. This transfer set is then fed to LLM in the process of fine-tuning.

As a disadvantage, this model slightly reduces NLP performance in high-resource languages. This is likely true since during the supervised fine-tuning process, the parameters change, which could have altered the performance in the high-resource language. Depending on the circumstances, this method may not bring the desired outcome, as the overall multilingual capabilities of the model may not be as high.

3.2.2 Parameter Efficient Fine-tuning (PEFT) (Ding et al., 2023)

PEFT is a novel approach to address the problems of increasing storage size, which escalates the computational and memory cost of LLMs. It focuses on tuning a

limited number of parameters, and modularity–meaning one LLM can be used for multiple specific tasks. It is based on the traditional fine-tuning method to modify the parameters to enhance performance.

3.2.3 Adapters (Houlsby et al., 2019)

Recognizing the issue of increasing required storage size, Neil et al proposed a method in 2019 that takes a novel approach to enhance the practicality of fine-tuning. This method inserts new layers in the neural networks called "Adapters" which are initialized at random. This occurs between every layer in the Transformer, which means that during tuning, the model creates a new learnable vector that guides to more accurate output. After tuning, and during actual interference, the model uses these new additional learned vectors to attain highly accurate output.

In adapter tuning, all of the old parameters are frozen–which means the weights and biases do not change– and only the parameters from the Adapters are tuned (~4% of the total parameters). This reduces the computational cost significantly, considering average of 1B tokens, updating 4% only cost approximately $200. The adapter's challenge is to create an effective layer architecture for the module.

Adapters have perfect memory of task-specific parameters from the model before tuning. The parameters are frozen, and instead, the new Adapters (new modules added between layers) are tuned. Therefore, the original task-specific content generated by the model does not get influenced as much as it does in full fine-tuning, mainly because of how only a few parameters are modified/added in total. For example, if the model was originally designed for "English to French" translation, and tuned for "English to Bengali", the new model will perform much better at "English to French" as well, if Adapters are used. This means that Adapters can enhance performance in low-resource languages while keeping high performance in high-resource languages.

3.2.4 Prefix tuning (Li & Liang, 2021)

Similarly to Adapters, prefix tuning is a parameter-efficient fine-tuning inspired by prompting. It freezes the model's parameters to minimize the increase in model

storage. Unlike Adapters, prefix tuning will add a task-specific vector to each layer in the Transformer to guide the original LLM to a more precise output. These vectors/parameters are prepended in front of every attention block in the Transformer. To prefix tune a GPT4.1 model, it will only cost about $7.50, significantly reducing the need for large investments.

	Adapter	Prefix tuning	Prompt tuning
Percentage of the total number of parameters to modify	~4%	~0.1%	~0.01%
Approximated cost (Using 1B model of GPT4.1)	$200	$7.5	$0.75
Performance	Highest	Moderate	Lowest

Table 1. Comparison of each PEFT method against the percentage of the total number of parameters to modify, approximated cost, and Performance.

In this section, we will generically compare the differences of the two major methods: Prompt-based methods and Fine-tuning methods.

	Prompt-based methods	Fine-tuning methods
What does it do	Adds learnable vectors onto the original prompt without changing the internal model parameters	Re-adjusts parameters (weights&biases) of LLMs1
Parameters updated	Very few: prompt tuning will modify a small portion of the parameters	Many: Every time fine tuning occurs, all of the parameters are updated
Computational cost	Minimal	Typically high due to mass storage
Data Efficiency	Effective with minimal data (e.g zero-shot)	Requires a massive amount of data
Performance increase	Lower:	Higher: as parameters fully adjust, which will enable the model to capture slight nuances.
Domain-specific Performance	Limited improvement for highly specialized tasks	Superior at adapting precisely to specialized domains, significantly boosting accuracy and task-specific performance

Computational/storage cost and performance increase, both in the context of low-resource languages. Unlike high-resource languages, low-resource languages tend to lack publicly accessible data (e.g, massive dictionaries) and have less investment to build the model due to the lack of demand. Furthermore, only one model needs to be built and deployed to all of the people who want to use the MT

for low-resource languages. This section's comparison will be in the specific context of MT in low-resource languages.

In terms of computational and storage cost, the prompt-based method tend to have efficient outcomes, as observed in Table 1. Prompt tuning was the most cost-effective method in the PEFT, suggesting its high accessibility and practicality in this context of low-resource languages. However, other than SDRRL, which is quite cost-heavy (pushing the boundary up to US$7,500), all of the methods only cost around US$10 (Adapters go up to US$200). In this context of aiming to create only one model for multiple users, the can become less important than its performance increase.

Viewing the increase in performance, SDRRL is very likely to achieve the highest of all with Adapters next. However, as mentioned in section 3.1.1, some of the prompt-based methods perform higher at specific situations. For example, DiPMT could perform better MT with rare words in the target sentence, whereas RePP could perform better at longer target sentences. Nevertheless, to yield the highest performance, SDRRL is likely to produce the best outcome.

4. Discussions

4.1 Restatement of Key Findings

The comparative analysis presented in this review demonstrates that all methods discussed—prompt-based methods and fine-tuning methods—achieved improvements in performance for Machine Translation (MT) in low-resource languages, although with distinct strengths and limitations. Prompt-based methods such as DiPMT, RePP, and Prompt Tuning emerged as efficient and cost-effective solutions, especially in scenarios with limited data and budget constraints. DiPMT notably excels in translating sentences with rare vocabulary due to its reliance on dictionary-based hints, showing a mean BLEU score improvement of +4.8 across various languages. Conversely, RePP is more suited for idiomatic and lengthy sentences due to its reliance on comprehensive bilingual phrase databases. Prompt tuning further stood out due to its remarkable cost-effectiveness and minimal parameter adjustment. Fine-tuning methods, notably PEFT techniques like

Adapters and Prefix tuning, showed substantial performance improvements at a slightly higher cost, with SDRRL providing the highest gains in accuracy. These results emphasize that while each method is viable, the choice of strategy should align closely with the specific linguistic characteristics, resource availability, and budgetary considerations inherent to the target low-resource language context.

4.2 Implications and Significance

The findings presented here underscore meaningful advances in adapting Large Language Models to low-resource languages, opening promising pathways for broader AI accessibility. The evaluation highlights that, despite varying strengths across methods, Parameter Efficient Fine-Tuning (PEFT) methods—particularly Adapters and Prefix Tuning—strike an effective balance between performance, cost-efficiency, and ease of implementation. Their modular and parameter-minimal approaches significantly reduce computational overheads, enhancing their practicality for deployment by smaller enterprises or individual developers. Nonetheless, ongoing research should focus on creating standardized evaluation metrics and customizable frameworks to accommodate diverse linguistic structures and data availability. Recognizing these factors will further improve method selection and optimization, ultimately contributing to a more inclusive and effective AI landscape. By continuing to refine these adaptable methods, the research community can better serve diverse linguistic populations, fostering equitable technological benefits across previously underserved communities.

4.3 Limitations

Limitations include the absence of a standardized benchmark dataset for comprehensive comparison across different methods, potential biases introduced by the qualitative rather than quantitative assessment criteria, as well as inherent disparity across linguistic complexity amongst low-resource languages.

First, the comparison of the methods was done qualitatively. This is due to the fact that there are no fair, comprehensive tests to assess the performance of all of the methods, so different papers used distinct methods to assess their models.

Furthermore, the limitation is that within low-resource languages, the performance varies. For example, RePP may have performed extensively with English-German but may not have performed better in another low resource language. This is because even within low-resource languages, the amount of data and their distinct linguistic structure vary. This sheds light on specific language-focused models.

5. Bibliography

Ding, N., Qin, Y., Yang, G., Wei, F., Yang, Z., Su, Y., Hu, S., Chen, Y., Chan, C.-M., Chen, W., Yi, J., Zhao, W., Wang, X., Liu, Z., Zheng, H.-T., Chen, J., Liu, Y., Tang, J., Li, J., & Sun, M. (2023). Parameter-efficient fine-tuning of large-scale pre-trained language models. *Nature Machine Intelligence*, 5(3), 220–235. https://doi.org/10.1038/s42256-023-00626-4

Ghazvininejad, M., Gonen, H., & Zettlemoyer, L. (2023). *Dictionary-based Phrase-level Prompting of Large Language Models for Machine Translation* (No. arXiv:2302.07856). arXiv. https://doi.org/10.48550/arXiv.2302.07856

Houlsby, N., Giurgiu, A., Jastrzebski, S., Morrone, B., Laroussilhe, Q. de, Gesmundo, A., Attariyan, M., & Gelly, S. (2019). *Parameter-Efficient Transfer Learning for NLP* (No. arXiv:1902.00751). arXiv. https://doi.org/10.48550/arXiv.1902.00751

Lester, B., Al-Rfou, R., & Constant, N. (2021). *The Power of Scale for Parameter-Efficient Prompt Tuning* (No. arXiv:2104.08691). arXiv. https://doi.org/10.48550/arXiv.2104.08691

Li, X. L., & Liang, P. (2021). *Prefix-Tuning: Optimizing Continuous Prompts for*

Generation (No. arXiv:2101.00190). arXiv. https://doi.org/10.48550/arXiv.2101.00190

Liu, P., Yuan, W., Fu, J., Jiang, Z., Hayashi, H., & Neubig, G. (2021). *Pre-train, Prompt, and Predict: A Systematic Survey of Prompting Methods in Natural Language Processing* (No. arXiv:2107.13586). arXiv. https://doi.org/10.48550/arXiv.2107.13586

Ming, L., Zeng, B., Lyu, C., Shi, T., Zhao, Y., Yang, X., Liu, Y., Wang, Y., Xu, L., Liu, Y., Zhao, X., Wang, H., Liu, H., Zhou, H., Yin, H., Shang, Z., Li, H., Wang, L., Luo, W., & Zhang, K. (2024). *Marco-LLM: Bridging Languages via Massive Multilingual Training for Cross-Lingual Enhancement* (No. arXiv:2412.04003). arXiv. https://doi.org/10.48550/arXiv.2412.04003

Papineni, K., Roukos, S., Ward, T., & Zhu, W.-J. (2002). Bleu: A Method for Automatic Evaluation of Machine Translation. In P. Isabelle, E. Charniak, & D. Lin (Eds.), *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics* (pp. 311–318). Association for Computational Linguistics. https://doi.org/10.3115/1073083.1073135

(PDF) *Multilingual Large Language Models and Curse of Multilinguality*. (2025, April 28). ResearchGate. https://doi.org/10.48550/arXiv.2406.10602

(PDF) Multilingual Prompting in LLMs: Investigating the Accuracy and Performance. (2024). *ResearchGate*. https://doi.org/10.55041/IJSREM17694

Qin, L., Chen, Q., Feng, X., Wu, Y., Zhang, Y., Li, Y., Li, M., Che, W., & Yu, P. S. (2024).

Large Language Models Meet NLP: A Survey (No. arXiv:2405.12819). arXiv. https://doi.org/10.48550/arXiv.2405.12819

Shin, J., Tang, C., Mohati, T., Nayebi, M., Wang, S., & Hemmati, H. (2025). *Prompt Engineering or Fine-Tuning: An Empirical Assessment of LLMs for Code* (No. arXiv:2310.10508; Version 2). arXiv. https://doi.org/10.48550/arXiv.2310.10508

Su, T., Peng, X., Thillainathan, S., Guzmán, D., Ranathunga, S., & Lee, E.-S. A. (2024). *Unlocking Parameter-Efficient Fine-Tuning for Low-Resource Language Translation* (No. arXiv:2404.04212; Version 1). arXiv. https://doi.org/10.48550/arXiv.2404.04212

Sun, Z., Jiang, Q., Huang, S., Cao, J., Cheng, S., & Wang, M. (2022). *Zero-shot Domain Adaptation for Neural Machine Translation with Retrieved Phrase-level Prompts* (No. arXiv:2209.11409). arXiv. https://doi.org/10.48550/arXiv.2209.11409

Tian, K., Mitchell, E., Yao, H., Manning, C. D., & Finn, C. (2023). *Fine-tuning Language Models for Factuality* (No. arXiv:2311.08401). arXiv. https://doi.org/10.48550/arXiv.2311.08401

Zhang, Y., Wang, Y., Liu, Z., Wang, S., Wang, X., Li, P., Sun, M., & Liu, Y. (2024). *Enhancing Multilingual Capabilities of Large Language Models through Self-Distillation from Resource-Rich Languages* (No. arXiv:2402.12204). arXiv. https://doi.org/10.48550/arXiv.2402.12204

Review Report

Manuscript: Enhancing Machine Translation in Low-Resource Languages: A Comparative Review of Prompt-Based and Fine-Tuning Methods

Summary

The manuscript provides a structured comparative review of prompt-based versus fine-tuning approaches for enhancing machine translation in low-resource languages. It discusses methods such as DiPMT, RePP, prompt tuning, adapters, and self-distillation. The paper is well organized and covers relevant literature.

Major Comments

1. Clarity of Writing (Page 2, Lines 34–45): The introduction is conceptually strong but uses technical terminology densely. Simplifying key definitions (e.g., 'parameter-efficient fine-tuning') in plain language would improve accessibility.

2. Critical Evaluation vs. Summary (Page 3, Lines 46–62): Much of the paper reads as a summary of prior work. A stronger critical voice—pointing out why one method may be more practical for low-resource communities—would add originality.

3. Balance of Content (Page 4, Lines 20–34): The analysis of computational cost vs. performance is informative, but the conclusion leans toward PEFT. A clearer framework of 'when to use which' would help readers.

4. Limitations (Page 5, Lines 64–73): The limitations are acknowledged but could be expanded. For example, note that cost estimates are highly dependent on hardware and experimental context.

Minor Comments

- Consistency: Ensure 'Prompt-based' and 'Fine-tuning' are capitalized uniformly.

- Abstract (Page 1, Line 17): Consider rephrasing 'fix the imbalance' → 'address the imbalance' for tone.

- References: Avoid duplicated entries (Page 2, Lines 1–15).

Recommendation

Accept with Minor revision.

Review for “Enhancing Machine Translation in Low-Resource Languages: A Comparative Review of Prompt-Based and Fine-Tuning Methods”

In this comparative review, the authors compare two machine learning architectures that can employed for translation. It is an interesting study that discusses cutting-edge algorithmic developments in language models. The proposed methods have been outlined well and more specific approaches have been identified from the literature. As such, this is a useful and timely contribution to the existing literature offering some novel insights into the main features of these methodologies and helping interpret the strengths and weaknesses of each approach.

- The literature has been properly cited and a few important recent studies have been identified. The methodology employed during the literature review could have been presented more rigorously by specifying how the studies included were selected and whether other studies identified were not considered and for what reasons. Nevertheless, the authors show critical understanding of the methodologies they outline and have added some mechanistic depth on what cases one should be preferred over the other and why.
- Recent findings have been reported although I would suggest that they have not been integrated fully between different studies to develop the authors’ argument with clarity. Specifically, to make the rationale clearer, it would be useful to reduce the number of subsections used and present more concisely their interpretation by pooling evidence across studies. To this end,
- Sections that directly contrast the different methodologies (rather than listing the features of each separately) would be very informative and would make the narrative more concise.
- To this end, the inclusion of graphical illustrations would strengthen the research paper and make it more accessible to a larger audience. I would recommend adding relevant figures that compare architectures/features and performance directly. This would make the comparisons more intuitive.
- I would also recommend trimming the text overall and focusing on the main messages.
- In terms of style, at times the writing can be slightly too technical and repetitive. I would recommend polishing the narrative by removing descriptions that do not add more intuition.

Overall, a more “direct” comparison between the methods would make the paper of publication quality.

My overall recommendation is: Revise and resubmit (major revisions needed, acceptance not guaranteed)

Enhancing Machine Translation in Low-Resource Languages: A Comparative Review of Prompt-Based and Fine-Tuning Methods

Author: [redacted]

Abstract:

Large language models (LLMs) such as LLama, GPT, T5, and Alpaca have demonstrated capabilities in multilingual tasks with natural language processing (NLP). However, low-resource languages often lack data regarding tokens and linguistic representations, which are critical for effective training of the models. This limits the models' performance, particularly in translation, compared to high-resource language models, which benefit from large labeled datasets. There is clearly a need to address the imbalance of effectiveness of LLMs in low-resource languages. While many methods exist to address this problem, it is unclear which method is best. In this review, we evaluate methods recently developed that attempt to enhance the translation performance of LLMs in low-resource languages. Specifically, we focus on prompt-based and Fine-Tuning methods - methods that have been shown to significantly improve the performance of LLMs in low-resource languages. Among the methods reviewed, Parameter-Efficient Fine-Tuning (PEFT)—including Adapters and Prefix Tuning—demonstrated the best balance between performance and cost, making them the most practical for low-resource languages. However, traditional full Fine-Tuning methods, despite higher computational requirements, achieve greater performance improvements, emphasizing the importance of balancing performance gains with resource constraints. These findings highlight the importance of balancing architectural enhancement methods with culturally and quantity-limited data. We conducted a narrative review with a structured search of ACL Anthology, Google Scholar, and arXiv, with predefined selection criteria.

1. Introduction

Prior works of adapting LLMs to specific domains used full Fine-Tuning, which can be costly and lacks effectiveness, specifically for low-resource languages. Methods to overcome these issues/disadvantages have been proposed. However, with the current state, there are too many options to choose from.

The fast-developing technology of large language models (LLMs) like Llama, GPT, T5, and Alpaca has transformed how humans interact with machines through content creation and human-like dialogue (Qin et al., 2024). For example, generative LLMs can generate content from any user input, such as text, images, sound, etc. These models operate using transformer-based architectures and are trained on extensive text corpora to identify linguistic patterns and accurately predict subsequent tokens. This enables LLMs to generate natural and coherent texts within a very short time.

Benefits of LLMs are not equally distributed across the world. A major contributor to this is the disparity of performance across different languages. The majority of LLMs are trained on high-resource languages, primarily English, where large amounts of text data are easily accessible. Therefore, LLMs respond significantly better to English commands and prompts than to languages that have less publicly accessible corpora- 'low-resource' languages (Ming et al., 2024; *(PDF) Multilingual Large Language Models and Curse of Multilinguality*, 2025). Approximately, there are 3 billion people worldwide who speak low-resource languages, and the differences in performance and usability cause inequality in linguistic representation and opportunity. For example, LLMs trained only in English cannot serve as a highly useful tool when it comes to assisting local governments, indigenous communities. For example, in one evaluation of translation tasks, English prompts achieved an average BLEU score of 92.5%, while Hindi prompts scored only 7.5% ("(PDF) Multilingual Prompting in LLMs," 2024). BLEU is a standard metric that evaluates how closely machine translations match human translations. Higher scores mean better translation quality(Papineni et al., 2002).

Researchers have tackled this problem by using the multilingual pretraining method (such as Macro-LLM), which involves training LLMs on texts from a mixture of languages to facilitate knowledge transfer across different languages (Ming et al., 2024). This approach enhances the models' ability to understand and generate low-resource language texts by leveraging knowledge from high-resource languages. However, this method caused a regression of the model's ability to respond to high-resource languages (*(PDF) Multilingual Large Language Models and Curse of Multilinguality*, 2025). Additionally, since the multilingual pretraining method requires large amounts of data for training, the imbalance in data availability between high-resource and low-resource languages leads to uneven performance.

Recently, novel approaches have been developed to address this problem. Fine-Tuning methods are methods that involve a direct adjustment of parameters (weights and biases, i.e, the numbers it uses to make predictions). These include full Fine-Tuning (FT; changing all parameters) and parameter-efficient Fine-Tuning methods (PEFT), which only change a small fraction. For example, Adapters $\approx 4\%$, Prefix Tuning $\approx 0.1\%$, Prompt Tuning $\approx 0.01\%$.

Prompt-based methods do not change the model at all. Instead, they change the prompt (input text) by adding guiding words or phrases that steer the model toward the right answer. This includes Dictionary-based phrase-level Prompting Machine Training (DiPMT), Retrieved Phrase-level Prompts (RePP), and prompt tuning(Ghazvininejad et al., 2023; Sun et al., 2022; Lester et al., 2021).

This review makes three contributions. First, we specify a structured search methodology. Second, we point towards specific methods and extract comparable data items across studies, such as model size, language pairs, metrics, % of parameters used, and tokens processed. Third, we provide a practical decision framework for reference on when to use which methods.

2. Methodology

A literature review with a structured search was conducted to identify relevant research on methods enhancing the performance of large language models (LLMs) for low-resource languages.

Databases: Google Scholar, ACL Anthology, and arXiv

Keywords: “low-resource languages”, “Multilingual large language model”, “prompt-based methods”, “Word Type coverage”, “metrics”, “parameter-efficient Fine-Tuning.”

The papers reviewed were selected based on clearly defined criteria. Each paper needed to clearly focus on adapting large language models, empirically evaluate either prompt-based methods or Fine-Tuning methods, include detailed evaluations regarding performance, computational efficiency, and costs, be peer-reviewed, and be written in English in a text format. Our goals were to identify recent methods specifically aimed at enhancing MT performance in low-resource languages, critically evaluate these methods using criteria such as computational practicality, implementation cost, and effectiveness, and to recommend the most suitable methods for practical implementation, considering restricted data constraints and linguistic variations in low-resource languages. Below are data that were extracted from the papers:

- Model (size)
- Language paris
- Methodology (prompt-based; full FT; PEFT)
- Metrics
- % parameters tuned
- Tokens processed
- Costs if reported

In this review, prompt-based methods were defined as an approach taken with zero modification of the parameters of the original model, focusing on enhancing the prompts to feed into the model to yield better performance. Conversely, Fine-Tuning methods refer to approaches that directly modify the parameters of the original model, thereby refining its transformer architecture or neural network structure to produce more effective outputs.

3. Comparative Results and Analysis

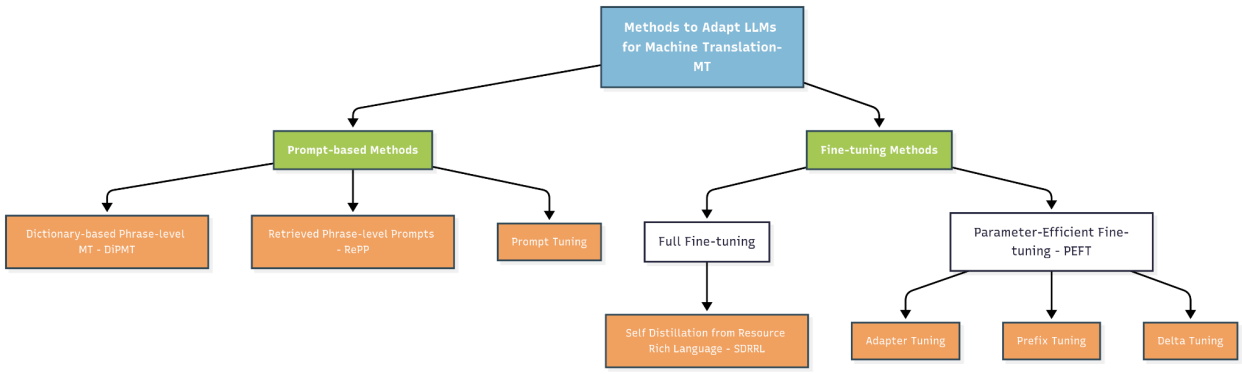

Figure 1. Represents the methods we have identified and how it is categorized. The methods are categorized into two large parts: Prompt-based Methods and Fine-Tuning Methods.

3.1 Prompt-based methods:

LLMs can be adapted to low-resource translation without retraining by modifying their inputs/prompts. Prompt-based methods require minimal computational cost and parameter updates (Liu et al., 2021; Shin et al., 2025). They differ mainly in how linguistic hints are provided to the model.

Initial language models, such as N-grams and Recurrent Neural Networks (RNNs) laid the foundation for prompt-based methods. Large-scale models such as GPT-3 enhanced the ability of prompt engineering to greater adaptability.

In this section, we compare the methods proposed by other researchers that were developed recently: Dictionary-based Phrase Level Prompting (DiPMT), Retrieval Phrase Level Prompting (RePP), and Prompt Tuning (Ghazvininejad et al., 2023; Lester et al., 2021; Sun et al., 2022).

3.1.1 DiPMT— Description and Key Result

Dictionary-based Phrase-level Machine Translation (DiPMT) was developed in 2023 by Meta AI (Ghazvininejad et al., 2023). DiPMT is a domain-specific method specialized in the task of Machine Translation. The core idea behind DiPMT is to enhance prompts using a bilingual dictionary to give hints for specific words, especially rare words, to narrow down the possibility of the interpretation for the LLMs.

In DiPMT, strings describing the meaning of the keywords (e.g. the word "pembuatan" means "creation") are appended to the original prompt. The addition of these strings expands the model's capability to accurately translate the original prompt. As a result of incorporating this method, the model showed an improvement in the BLEU compared to the baseline LLM output (which omits the dictionary hints from the prompt) with a mean increase average of +4.8, and a standard deviation of ±1.2 across 17 languages of the 20 tested. An example prompt in a DiPMT:

> Translate the following sentence to English: Ia melakukan pembuatan bel pintu dengan teknologi WiFi, katanya.
> In this context, the word "pembuatan" means "creation"; the word "bel" means "buzzer", "bell"; the word "pintu" means "door", "doors".
> The full translation to English is:

Figure 2. Example of a zero-shot prompt using DiPMT. Green words, which act as hints, are added to the original blue & red instructions.

This method highly depends on the type of coverage—Word Type Coverage and Word Token Coverage—of the dictionary for the target language. The Word Type Coverage of a dictionary represents the proportion of distinct words in the language that exist in the dictionary. A high Word Type Coverage means the dictionary is robust and covers diverse vocabulary. On the other hand, Word Token Coverage

considers the percentage of dictionaries that can be translated in real-world situations. Thus, small dictionaries can have high Token Coverage if they contain frequent common words in the language. This relationship underscores the importance of lexical coverage: when dictionaries include a wide variety of word types, models can better clarify meanings during translation, especially in underrepresented languages.

The study shows that DiPMT only performs extensively better when the Type Coverage is above a certain level (5 - 20%). Typical lexical coverage for low-resource languages is around 20 - 50%, which suggests that DiPMT is practically useful in real-life situations.

The performance of the method is also highly dependent on the hint that is provided. The prompts produced by the normal DiPMT method provide all available dictionary translations of words in the original prompt, creating multiple competing options that can be challenging for LLMs to determine the most contextually appropriate choice. The results from the paper show that the output from this method can sometimes be inaccurate since the model could potentially choose the wrong definition. Although this method can be highly accurate when "gold hints" –which are the correct translations of some key words in the original prompts– are provided, this may not be possible to attain for every case of translation. Thus, a method that would allow gold hints to be derived from dictionary hints will increase the effectiveness of DiPMT significantly; otherwise, the performance may be limited.
Overall, DiPMT is strongest for rare-word translation when a robust bilingual dictionary is available, but its success is tightly constrained by dictionary coverage.

3.1.2 RePP— Description and Key Result

Retrieved Phrase-level Prompts (RePP) enhances LLM's outputs by retrieving relevant phrase-level data from a bilingual dictionary database and combining this with the input before processing in the LLM (Sun et al., 2022). Additional precision from the database clarifies meanings for the LLM to process, and using phrase-level data limits the possibility of interpretations compared to sentence-level data.

RePP enhances the prompt in three automated processes: segmenting the input into phrases, retrieving aligned bilingual phrases from a pre-built database, and concatenating them before translation. RePP specially integrates phrase-level bilingual data without retraining the model, improving clarity and cross-lingual alignment.

In benchmark tests on English-German and German-English translation, RePP improved mean BLEU scores by +6.2 compared to baseline models, performing slightly below FFT. Because the bilingual phrases database combines multiple dictionaries, its Word Type Coverage and Token Coverage are both high, making RePP more stable than DiPMT, which relies only on one dictionary.

Unlike FT, RePP requires only a single base model for multiple domains, since each input prompt is augmented with phrase-level bilingual data. This adaptability makes it especially robust for low-resource languages, where corpora (large text data) and funding (for advanced models) are scarce.

3.1.3 Prompt Tuning— Description and Key Result

Prompt tuning is a parameter-efficient strategy that freezes the pretrained parameters and inserts a small, learnable prompt vector into the model's embedding layer (Lester et al., 2021).

The vector—a sequence of virtual tokens prepended to the input—guides the model toward task-specific outputs without altering its original parameters.

In the presented paper, Lester et al. analyzed prompt tuning's performance using GLUE and SuperGLUE benchmarks, and the model they used was T-5 XXL. This method matched FFT performance (approximately 89.3 accuracy) while updating only approximately 0.01% of parameters.

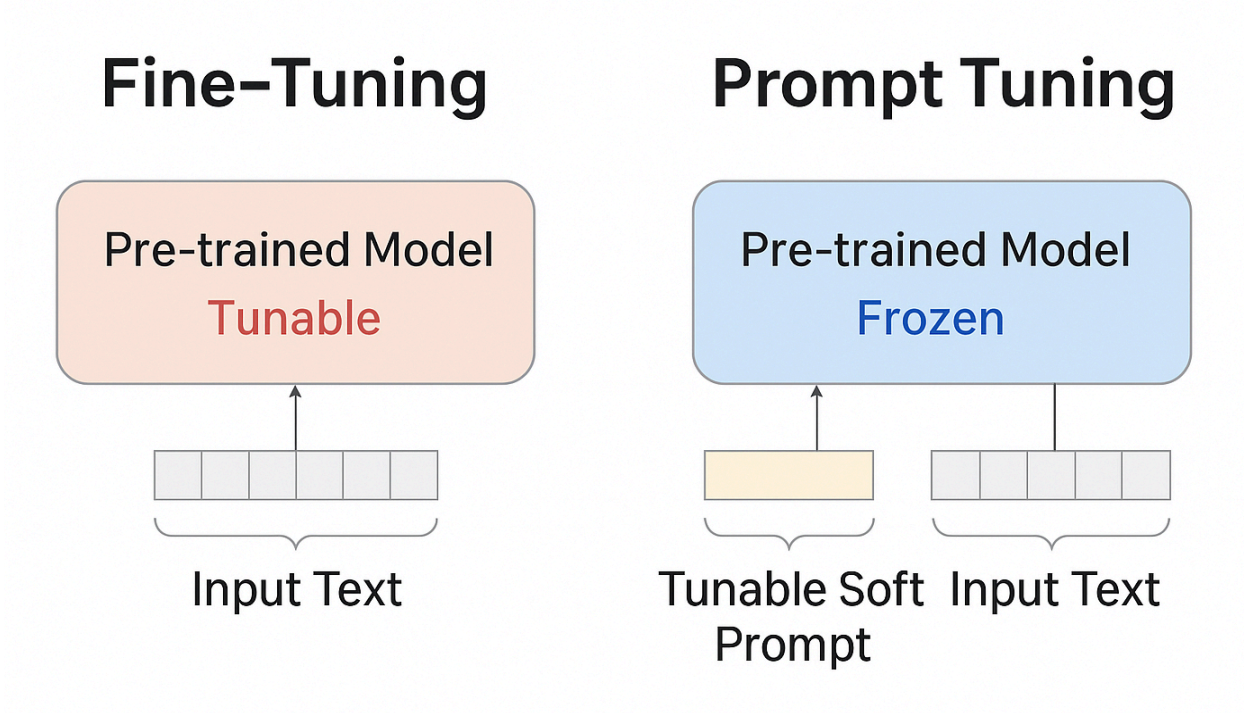

Figure 3. Conceptual difference between Fine-Tuning and Prompt Tuning.

Because only the prompt vectors are developed, the computational cost is negligible. Freezing the original parameters also preserves high-resource-language performance, an advantage in multilingual machine translation, where maintaining English or other prominent language accuracy while adding low-resources translation capability is essential.

3.1.4 Comparison and Trends among Prompt-Based Methods

Table 1. Performance and characteristics of prompt-based methods (DiPMT, RePP, and Prompt Tuning) across low-resource machine translation tasks. (BLEU = translation quality metric; higher = better.)

Method	% of parameters modified	Data or resource needed	Relative Cost	MT performance (BLEU, GLUE)	Best use cases	Risk/cost

DiPMT	0%	A bilingual dictionary	Low; minimal	Average +4.8 BLEU; especially effective in out-of-domai n situation	Sentences with rare, uncommon words; scenarios with dictionaries with high Word Coverage	If the dictionary lists multiple translations, it can lead to lower accuracy; quality depends on the dictionary's Word Coverage
RePP	0%	A phrase pair database made from multiple dictionaries	Low	+6.2 BLEU on En-De	Fast domain adaptation when phrases are available; sentences with idiomatic expressions	Building a phrase pair database; highly relies on the accuracy of the phrase pair
Prompt Tuning	~0.01%	None	Low-moderat e	Achieving close to FFT in the GLEU score	When protecting the main resource-rich language's performance	Underperfor ms compared to DiPMT and RePP

Methods that bring the right translation words improve the performance the most. DiPMT scales with dictionary Word Coverage; when it is below 20%, it faces a significant shrinkage in improvements. RePP is more consistent as it uses multiple bilingual dictionaries to create a phrase pair database, but requires more time and cost to build. Pure prompting methods keep costs near zero, but wrong or noisy hints can mislead the model and lower the performance. Thus, the quality of the external source highly matters. Prompt Tuning changes almost nothing, so it is ideal when you must keep multilingual breadth while adding a task, but it usually lacks the precision compared to DiPMT and RePP.

3.2 Fine-Tuning Methods:

LLMs can also be adapted to low-resource language translation by retraining parts of their internal parameter— a.k.a Fine-Tuning. Unlike Prompt-based methods, which modify the inputs, Fine-Tuning methods directly adjust the model's behavior (Ding et al., 2023; Houlsby et al., 2019; Li & Liang, 2021)

Early approaches to Fine-Tuning retrained all parameters—a.k.a Full Fine-Tuning (FFT). This method achieved strong task specialization at the expense of high cost and scalability. To reduce the burden, methods like Self-Distillation from Resource-Rich Language (SDRRL) and Parameter-Efficient Fine-Tuning (PEFT) has emerged, which update only a fraction of parameters while keeping most of the model frozen.

Parameter Efficient Fine-Tuning (PEFT) (Ding et al., 2023)is a novel approach to address the problems of increasing storage size, which escalates the computational and memory cost of LLMs. It focuses on tuning a limited number of parameters and modularity–meaning one LLM can be used for multiple specific tasks. PEFT we review are:

1. Adapters
2. Prefix Tuning

3.2.1 SDRRL— Description and Key Result

Self-Distillation from Resource-Rich Languages (SDRRL) (Zhang et al., 2024) mitigates data scarcity by transferring patterns from high-resource to low-resource languages.

In SDRRL, the model first generates an exemplar response in a high-resource language (e.g English), assuming it is accurate. Then, it translates them to the target low-resource language, and compiles these pairs into a data set used for supervised Fine-Tuning. This process of cross-lingual transfer boosts low-resource performance but can slightly reduce the accuracy in the high-resource language because some parameters are overwritten during supervised Fine-Tuning. Though, Zhang et al. (2024) observed that SDRRL improved BLEU by 2-5 points across 14 languages while largely preserving accuracy in English and French

SDRRL creates its own data by translating high-quality output, which effectively removes the dependence on existing bilingual corpora. Transferring sentence-level grammtaical knowledge from high-resource languages rather than individual lexical pairs, the model learns linguistic structures—such as word order, tense, and dependencies—that dictionaries may not capture. The deeper grammatical learning advantages by fluent translation in low-resource settings.

3.2.2 Adapters— Description and Key Result

Recognizing the issue of increasing required storage size, Houlsby et al. proposed a method in 2019 that takes a novel approach to enhance the practicality of Fine-Tuning (Houlsby et al., 2019).

In this method Adapters—tiny trainable layers— are added between frozen layers of the Transformer model (Vaswani et al., 2017). During Fine-Tuning, only these adapter layers are trained which makes the process faster and cost-effective. Typically, about 4% parameters are tuned while 96% remain fixed.

When training 1 billion tokens, adapter Fine-Tuning costs roughly $200 compared to thousands for FFT. Adapters improved F1/BLEU by roughly 12% across eight low-resource languages (Gurgurov et al., 2024).

The architecture allows the model to remember what is already known from high-resource language, allowing cross-lingual transfer, which can efficiently enhance the performance of low-resource languages while maintaining performance in high-resource languages.

Because Adapters can be added or removed like plug-ins, one shared model can hold multiple language-specific adapters at once, each trained separately and loaded only when needed, ensuring the model's robustness.

3.2.3 Prefix tuning— Description and Key Result

Similar to Adapters, prefix tuning is a parameter-efficient Fine-Tuning inspired by prompting (Li & Liang, 2021). It freezes the model's parameters to minimize the increase in model storage. Unlike Adapters, prefix tuning will add a task-specific vector to each layer in the Transformer to guide the original LLM to a more precise output. These vectors/parameters are prepended in front of every attention block in the Transformer. To prefix tune a GPT4.1 model, it will only cost about $7.50, significantly reducing the need for large investments.

3.2.4 Comparison and Trends among Fine-Tuning Methods

Table 2. Comparison of Fine-Tuning strategies (Full Fine-Tuning, SDRRL, Adapters, and Prefix Tuning) on efficiency, data use, and translation quality.

Method	% of parameters modified	Data or resource needed	Relative cost	MT performance (BLEU, GLUE)	Best use cases	Risk/cost
FFT	100%	Large parallel corpora	Very high (~$7500 per model)	~+12 BLEU in domain-specific tasks	With a high budget and data, need for maximum stability on one domain	Forgetting other languages/domains; expensive
SDRRL	>50%	Output from high-resource languages	Medium-high	~+5 BLEU across 14 languages from English/French models	Lacking low-resource language data; have access to high-resource language models	Some loss to high-resource language performance; though less than FFT, costs heavily
Adapter	~4%	Moderate bilingual data	Low-medium	Near FFT accuracy in most low-resource	High accuracy per cost; need for multiple	Managing many adapters; some tasks

				language settings; highest among PEFT	domains	may perform lower than FFT
Prefix tuning	~0.1%	Moderate bilingual data	Low	Near FFT accuracy in most low-resource language settings	Moderate accuracy per cost; need for multiple domains; need for rapid development	May not have reach Adapter or FFT performance.

The trend in Fine-Tuning models shows that the performance strongly correlates with the % of parameters modified. Updating more tends to give more BLEU—FFT at the top, PEFT in the middle—but with higher cost and lower performance in the original high-resource language.

Adapters and Prefix-tuning gains nearly as much as FFT for a fraction of the cost. SDRRL makes its own training pairs by translating high-resource language output first; this improves BLEU scores for low-resource language when parallel data is scarce.

3.3 Cross Category Evaluation

Table 3. Comparative overview of adaptation methods for low-resource translation

Method Type	Example Methods	Parameters Modified	Data Efficiency	Relative Cost	BLEU Performance	Domain-specific Performance	Strengths	Limitations
Prompt-based	DiPMT, RePP, Prompt Tuning	0-0.01%	Effective with minimal data (e.g zero-shot)	Lowest (<$1 to $10 per run)	+4-6 BLEU gain over baseline	Limited improvement for highly specialized tasks	Zero retraining; fast deployment; high accessibility	Lowest accuracy; depends on lexical coverage
Fine-Tuning methods	PEFT (Adapters, Prefix Tuning), SDRRL	0.1-50%	Moderate data requirement (e.g bilingual corpora)	Low-Medium (~$7.50 - $200 per billion tokens)	+8-13	Superior at adapting to specialized domains; captures grammatical and semantic	Reusable modules; preserves primary language competence; high accuracy	Medium-High accuracy, architectural complexity

						structure	per cost	
Full Fine-Tuning (FFT)	Traditional FT	100%	Requires large-scale parallel datasets	Very High (~$7500 per model)	+10-12	Highest precision and stability on task-specific domains	Full adaptation; maximal accuracy	Costly, prone to forgetting

The difference in machine-translation performance across methods primarily arises from the % of parameter modification. SDRRL retrains >50% of the model, achieving the greatest improvement (~+13 BLEU) with medium-high computational costs. PEFTs, like Adapters, achieve nearly the same improvements as FFT (~+12 BLEU) while altering only small portions of the model's parameters (~4%), allowing domain-specific adaptation while retaining pretrained knowledge. By contrast, Prompt-based methods, like Prompt Tuning, modify only ~0.01% of parameters, proposing a quicker adaptation, but with less significant improvement (~+6 BLEU).

Differences in data efficiency also influence the translation quality. Prompt-based methods perform effectively with minimal data, operating under zero- or few-shot environments without bilingual corpora. PEFT methods, such as Prefix and Adapter, require moderate training data. Consequently training data requirements increase in proportion to the percentage of parameters updated.

The relative cost is proportional to the % parameters modified. Prompt-based and Prefix-Tuning methods operate at under $10 per modification, while Adapters require ~$200 per 1 B tokens (Houlsby et al., 2019). SDRRL and FFT require the highest costs, ranging from $2000 - $7500 per model(Ding et al., 2023).

These quantitative comparisons establish the relative strengths of each method across performance and cost dimensions. The broader implications of these trade-offs for low-resource language settings are discussed in Section 4.

4. Discussions

4.1 Restatement of Key Findings

Evaluation results demonstrate clear proportionality between parameter modification, computational costs, and MT performance. Methods updating fewer than 1% of parameters yield modest BLEU improvements (~+6), whereas those up to ~50% modification achieved near-FFT accuracy (~+13) at much lower cost. FFT consistently achieves the highest accuracy but with the greatest cost burden (~$7500 per model).

This trend reflects each method's mechanism. Prompt-based methods rely on lexical or phrase, bit by bit translation rather than direct adjustments of the parameters, which explains their speedy development but

with limited domain precision. PEFT methods introduce lightweight trainable modules to enhance the accuracy of domain-specific tasks while preserving pre-trained linguistic knowledge in resource-rich languages. SDRRL's cross-lingual self-distillation enables transfer of linguistic knowledge from resource-rich languages, explaining the marginal lead over other Fine-Tuning methods.

4.2 Implications and Significance

The findings presented here underscore meaningful advances in adapting Large Language Models to low-resource languages, opening promising pathways for broader AI accessibility. The evaluation highlights that, despite varying strengths across methods, Parameter Efficient Fine-Tuning (PEFT) methods—particularly Adapters and Prefix Tuning—strike an effective balance between performance, cost-efficiency, and ease of implementation. Their modular and parameter-minimal approaches significantly reduce computational overheads, enhancing their practicality for deployment by smaller enterprises or individual developers. Nonetheless, ongoing research should focus on creating standardized evaluation metrics and customizable frameworks to accommodate diverse linguistic structures and data availability. Recognizing these factors will further improve method selection and optimization, ultimately contributing to a more inclusive and effective AI landscape. By continuing to refine these adaptable methods, the research community can better serve diverse linguistic populations, fostering equitable technological benefits across previously underserved communities.

4.3 Limitations

Limitations include the absence of a standardized benchmark dataset for comprehensive comparison across different methods, potential biases introduced by the qualitative rather than quantitative assessment criteria, as well as inherent disparity across linguistic complexity amongst low-resource languages.

First, the comparison of the methods was done qualitatively. This is due to the fact that there are no fair, comprehensive tests to assess the performance of all of the methods, so different papers used distinct methods to assess their models.

Furthermore, the limitation is that within low-resource languages, the performance varies. For example, RePP may have performed extensively with English-German but may not have performed better in another low-resource language. This is because even within low-resource languages, the amount of data and their distinct linguistic structure vary. This sheds light on specific language-focused models.

5. References

Ding, N., Qin, Y., Yang, G., Wei, F., Yang, Z., Su, Y., Hu, S., Chen, Y., Chan, C.-M., Chen, W., Yi, J.,

Zhao, W., Wang, X., Liu, Z., Zheng, H.-T., Chen, J., Liu, Y., Tang, J., Li, J., & Sun, M. (2023).

Parameter-efficient Fine-Tuning of large-scale pre-trained language models. *Nature Machine Intelligence*, *5*(3), 220–235. https://doi.org/10.1038/s42256-023-00626-4

Ghazvininejad, M., Gonen, H., & Zettlemoyer, L. (2023). *Dictionary-based Phrase-level Prompting of Large Language Models for Machine Translation* (No. arXiv:2302.07856). arXiv. https://doi.org/10.48550/arXiv.2302.07856

Gurgurov, D., Hartmann, M., & Ostermann, S. (2024). Adapting Multilingual LLMs to Low-Resource Languages with Knowledge Graphs via Adapters. *Proceedings of the 1st Workshop on Knowledge Graphs and Large Language Models (KaLLM 2024)*, 63–74. https://doi.org/10.18653/v1/2024.kallm-1.7

Houlsby, N., Giurgiu, A., Jastrzebski, S., Morrone, B., Laroussilhe, Q. de, Gesmundo, A., Attariyan, M., & Gelly, S. (2019). *Parameter-Efficient Transfer Learning for NLP* (No. arXiv:1902.00751). arXiv. https://doi.org/10.48550/arXiv.1902.00751

Lester, B., Al-Rfou, R., & Constant, N. (2021). *The Power of Scale for Parameter-Efficient Prompt Tuning* (No. arXiv:2104.08691). arXiv. https://doi.org/10.48550/arXiv.2104.08691

Li, X. L., & Liang, P. (2021). *Prefix-Tuning: Optimizing Continuous Prompts for Generation* (No. arXiv:2101.00190). arXiv. https://doi.org/10.48550/arXiv.2101.00190

Liu, P., Yuan, W., Fu, J., Jiang, Z., Hayashi, H., & Neubig, G. (2021). *Pre-train, Prompt, and Predict: A Systematic Survey of Prompting Methods in Natural Language Processing* (No. arXiv:2107.13586). arXiv. https://doi.org/10.48550/arXiv.2107.13586

Ming, L., Zeng, B., Lyu, C., Shi, T., Zhao, Y., Yang, X., Liu, Y., Wang, Y., Xu, L., Liu, Y., Zhao, X., Wang, H., Liu, H., Zhou, H., Yin, H., Shang, Z., Li, H., Wang, L., Luo, W., & Zhang, K. (2024). *Marco-LLM: Bridging Languages via Massive Multilingual Training for Cross-Lingual Enhancement* (No. arXiv:2412.04003). arXiv. https://doi.org/10.48550/arXiv.2412.04003

Papineni, K., Roukos, S., Ward, T., & Zhu, W.-J. (2002). Bleu: A Method for Automatic Evaluation of Machine Translation. In P. Isabelle, E. Charniak, & D. Lin (Eds.), *Proceedings of the 40th Annual*

Meeting of the Association for Computational Linguistics (pp. 311–318). Association for Computational Linguistics. https://doi.org/10.3115/1073083.1073135

(PDF) Multilingual Large Language Models and Curse of Multilinguality. (2025, April 28). ResearchGate. https://doi.org/10.48550/arXiv.2406.10602

(PDF) Multilingual Prompting in LLMs: Investigating the Accuracy and Performance. (2024). *ResearchGate*. https://doi.org/10.55041/IJSREM17694

Qin, L., Chen, Q., Feng, X., Wu, Y., Zhang, Y., Li, Y., Li, M., Che, W., & Yu, P. S. (2024). *Large Language Models Meet NLP: A Survey* (No. arXiv:2405.12819). arXiv. https://doi.org/10.48550/arXiv.2405.12819

Shin, J., Tang, C., Mohati, T., Nayebi, M., Wang, S., & Hemmati, H. (2025). *Prompt Engineering or Fine-Tuning: An Empirical Assessment of LLMs for Code* (No. arXiv:2310.10508; Version 2). arXiv. https://doi.org/10.48550/arXiv.2310.10508

Sun, Z., Jiang, Q., Huang, S., Cao, J., Cheng, S., & Wang, M. (2022). *Zero-shot Domain Adaptation for Neural Machine Translation with Retrieved Phrase-level Prompts* (No. arXiv:2209.11409). arXiv. https://doi.org/10.48550/arXiv.2209.11409

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł. ukasz, & Polosukhin, I. (2017). Attention is All you Need. *Advances in Neural Information Processing Systems*, *30*. https://proceedings.neurips.cc/paper_files/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html

Zhang, Y., Wang, Y., Liu, Z., Wang, S., Wang, X., Li, P., Sun, M., & Liu, Y. (2024). *Enhancing Multilingual Capabilities of Large Language Models through Self-Distillation from Resource-Rich Languages* (No. arXiv:2402.12204). arXiv. https://doi.org/10.48550/arXiv.2402.12204

Reviewer 1

1. **Comment:** The introduction is conceptually strong but uses technical terminology densely. Simplifying key definitions (e.g., “parameter-efficient fine-tuning”) in plain language would improve accessibility.
   Response: Simplified the definition of parameter-efficient fine-tuning and added numeric examples (“Adapters ≈ 4%, Prefix ≈ 0.1%, Prompt ≈ 0.01%”) in §1 and §3.1 to make the meaning clear to general readers.

2. **Comment:** Much of the paper reads as a summary of prior work. A stronger critical voice—pointing out why one method may be more practical for low-resource communities—would add originality.
   Response: Added explicit comparison and interpretation paragraphs in §3.1.4 and §3.2.4 explaining why particular methods are practical for low-resource contexts, and emphasized these trends in §3.3.

3. **Comment:** The analysis of computational cost vs. performance is informative, but the conclusion leans toward PEFT. A clearer framework of “when to use which” would help readers.
   Response: Inserted new tables (Tables 1–3) that include Best use cases and Risk/Cost columns, providing a clear decision framework on when each method is most suitable.

4. **Comment:** The limitations are acknowledged but could be expanded. For example, note that cost estimates are highly dependent on hardware and experimental context.
   Response: Expanded §4.3 to discuss hardware-dependence, contextual variability in cost estimation, and qualitative comparison limitations.

Minor Comments

5. **Comment:** Consistency: Ensure ‘Prompt-based’ and ‘Fine-tuning’ are capitalized uniformly.
   Response: Standardized capitalization throughout.

6. **Comment:** Abstract (Page 1, Line 17): Consider rephrasing ‘fix the imbalance’ → ‘address the imbalance’ for tone.
   Response: Changed to “address the imbalance.”
7. **Comment:** References: Avoid duplicated entries (Page 2, Lines 1–15).
   Response: Removed duplicate citations in the bibliography.

Reviewer 2

1. **Comment:** The methodology employed during the literature review could have been presented more rigorously by specifying how the studies included were selected and whether other studies identified were not considered and for what reasons.
   Response: Rewrote §2 (Methodology) to detail search databases, keywords, inclusion criteria, and extracted data fields.
2. **Comment:** Recent findings have been reported although I would suggest that they have not been integrated fully between different studies to develop the authors’ argument with clarity. Specifically, to make the rationale clearer, it would be useful to reduce the number of subsections used and present more concisely their interpretation by pooling evidence across studies.
   Response: Merged and condensed sections into three synthesis parts (§3.1.4, §3.2.4, §3.3) that pool evidence and reduce fragmentation.
3. **Comment:** Sections that directly contrast the different methodologies (rather than listing the features of each separately) would be very informative and would make the narrative more concise.
   Response: Added cross-category comparison section (§3.3) and comparison tables explicitly contrasting Prompt-based vs Fine-tuning methods.
4. **Comment:** The inclusion of graphical illustrations would strengthen the research paper and make it more accessible to a larger audience. I would recommend adding relevant figures that compare architectures/features and performance directly.
   Response: Inserted new Figure 1 (method taxonomy) and Figure 3 (fine-tuning vs prompt-tuning), plus tables summarizing comparative performance and cost.
5. **Comment:** I would also recommend trimming the text overall and focusing on the main messages.

Response: I deleted repeated parts and made the Results and Discussion sections shorter so that the main points are easier to understand.

6. **Comment:** In terms of style, at times the writing can be slightly too technical and repetitive. I would recommend polishing the narrative by removing descriptions that do not add more intuition.
   Response: Rewrote introductory sentences in plain language (e.g., “LLMs can be adapted to low-resource translation without retraining…”) and simplified technical phrasing throughout.
7. **Comment:** Overall, a more direct comparison between the methods would make the paper of publication quality.
   Response: Added explicit “trend” paragraphs in §3.3 and a decision framework table summarizing method strengths, costs, and suitability.

Summary of Changes Across the Paper

– Added structured methodology section.
– Integrated findings into concise comparison subsections.
– Added new figures and comparison tables.
– Simplified tone and reduced technical repetition.
– Expanded limitations to clarify contextual variability and qualitative nature.

The authors have improved the narrative of the paper considerably during this revision. In particular, the comparison between the two approaches is now more direct and insightful. I would suggest merging a few short paragraphs into bigger ones with one main point.