MLM: Multi-linguistic LoRA Merging

Jung Lee[*]
Sogang University
junglee97@sogang.ac.kr

Taero Kim[*]
Yonsei University
taero.kim@yonsei.ac.kr

Nikhil Verma[†]
LG Toronto AI Research lab
nikhil.verma@lge.com

Abstract

Large Language Models (LLMs) often struggle to generalize across languages, exhibiting strong task performance in high-resource settings but substantial degradation in low-resource ones. This performance gap arises not only from task-specific data scarcity, but also from the language imbalance embedded in pre-training. In this work, we propose Multi-Linguistic LoRA Merging (MLM), a modular fine-tuning framework that decouples task and language adaptation into two independently trained LoRA-based adapters: a Task Adapter (TA) trained on a high-resource language, and a Language Adapter (LA) trained on the target low-resource language while keeping both the TA and base model frozen. The LA serves as a linguistic bridge, aligning language-specific representations to the task logic encoded in the TA. The two adapters are then combined through simple parameter-space interpolation, followed by a lightweight post-merging alignment stage to refine their interaction. This design enables highly sample-efficient training, requiring only a single-epoch update for each new language adapter, while preserving strong task knowledge from the TA. We evaluate MLM on the MMLU-ProX benchmark across multiple train/test splits and model sizes (LLaMA-3.2 1B and 3B), demonstrating consistent improvements over strong baselines in both Spanish and Hindi transfer. Our results highlight that modular, decoupled adaptation provides an effective and scalable recipe for efficient multilingual fine-tuning.

1 Introduction

Large Language Models (LLMs) have rapidly advanced the state of natural language processing, demonstrating strong performance in reasoning, generation, and general-purpose task understanding (Brown et al., 2020; Achiam et al., 2023). As these models continue to be deployed in global applications—from education and healthcare to government and customer service—their ability to operate effectively across multiple languages becomes increasingly important (Liu and Fu, 2024; Zhu et al., 2024). Multilingual capability is not merely a desirable feature, but a core requirement for ensuring inclusiveness and fairness in real-world AI systems (Ramesh et al., 2023).

Despite this growing demand, LLMs perform strongly on high-resource languages like English, but often fail to generalize to low-resource languages, even when evaluated on identical tasks (Muennighoff et al., 2022; de Wynter et al., 2025; Huang et al., 2025). This disparity, as highlighted in multilingual benchmarks such as MMLU-ProX (Xuan et al., 2025), reflects the base model's uneven language capabilities rather than simply differences in fine-tuning data (Ahuja et al., 2023). As LLMs are increasingly deployed in multilingual contexts, bridging this performance gap is essential for equitable and reliable AI systems. To this end, there is a growing need for fine-tuning strategies that can effectively compensate for pre-trained disparities and transfer task knowledge from high-resource to low-resource languages in a modular and efficient manner.

[*]Equal Contribution
[†]Corresponding Author

39th Conference on Neural Information Processing Systems (NeurIPS 2025) Workshop: Efficient Reasoning.

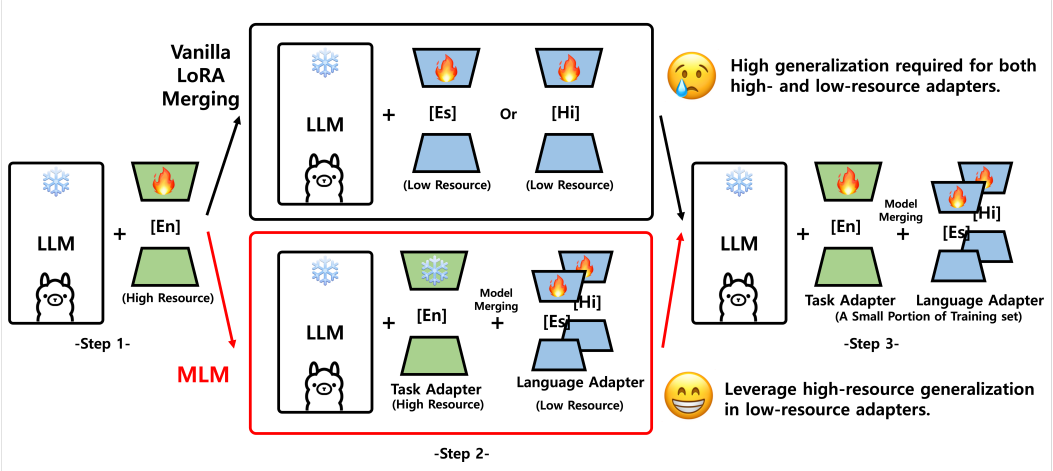

Figure 1: `En` stands for English, `Es` for Spanish, and `Hi` for Hindi. Step 1. Train a LoRA adapter using a high-resource language such as `En`. In MLM, this adapter is referred to as the TA. Step 2. In vanilla LoRA merging approaches, separate adapters should be trained for each low-resource language. In contrast, MLM incorporates the pre-trained TA into the training process of low-resource adapters, enabling the resulting LAs to effectively capture linguistic information. The integration between TA and LAs can be realized using various LoRA merging techniques. Step 3. The trained adapters are merged using different LoRA merging strategies, and, if necessary, an additional alignment step is applied using a subset of training data. In MLM, this step can be interpreted as refining the connection between the task knowledge encoded in the TA and the linguistic knowledge captured by the LAs.

However, despite rapid progress in scaling model capacity and pretraining corpora, multilingual robustness remains limited. Recent studies have shown that even instruction-tuned models struggle to retain accuracy on low-resource languages, especially when exposed to complex reasoning tasks (Chai et al., 2025; Lai and Nissim, 2024). Moreover, the prevailing "English-centric" pre-training paradigm inherently favors languages with abundant or typographically similar resources, leading to biased representations that transfer poorly to morphologically rich or syntactically divergent languages (Pires et al., 2019). These structural limitations underscore the importance of designing multilingual adaptation methods that explicitly address disparities between languages, rather than relying solely on scale.

Building on these challenges, recent research has explored LoRA-based adapter merging as a modular and parameter-efficient fine-tuning strategy (Hu et al., 2022). For instance, LoRA Soups (Prabhakar et al., 2024) and LoRA-LEGO (Zhao et al., 2025) introduce techniques to merge adapters trained on diverse tasks, allowing for improved generalization without retraining the entire model. While these approaches primarily target task compositionality rather than multilingual disparity, their modular framework suggests potential extensions to cross-lingual settings. In contrast, AdaMergeX (Zhao et al., 2024) explicitly addresses multilingual adaptation by decoupling task and language information into separate adapters and merging them through a structure-aware process. However, AdaMergeX requires auxiliary reference tasks and involves complex merging procedures, which can introduce additional overhead in practical deployment.

In this work, we propose *Multi-Linguistic LoRA Merging (MLM)*, a simple yet effective framework for multilingual adaptation. MLM introduces two modular adapters: a Task Adapter (TA) trained on a high-resource language to capture task semantics, and a Language Adapter (LA) trained on a low-resource language while keeping both the TA and the base model frozen. This separation ensures that the LA focuses solely on bridging linguistic gaps while preserving task knowledge from the TA. By decoupling task and language adaptation, MLM avoids redundant retraining, supports plug-and-play extensibility for new languages, and enables sample-efficient training. Concretely, the TA is trained once for a few full epochs, while each new LA requires only a single-epoch update followed by a lightweight alignment stage. Unlike prior methods that rely on complex merging procedures, MLM employs a simple parameter-space interpolation to combine task and language adapters, followed by a brief post-merging alignment step that ensures smooth integration across languages.

We evaluate MLM on MMLU-ProX, where identical question-answering (QA) tasks are translated across languages to isolate language effects. In addition, we assess the robustness of MLM across varying model sizes and train/test splits. MLM consistently improves performance on low-resource languages, demonstrating its modularity, efficiency, and effectiveness in multilingual transfer. We confirm this on two LLaMA-3.2 backbones (1B and 3B parameters), showing that MLM scales to larger models without sacrificing accuracy, while retaining efficiency gains that become increasingly pronounced as the number of target languages grows.

Our Contributions.

- **Modular Decoupling:** We introduce the *Multi-Linguistic LoRA Merging* (MLM) framework that *explicitly separates* task and language knowledge into two lightweight adapters, enabling plug-and-play multilingual reuse.
- **Adapter-Level Merging and Alignment:** We employ a simple parameter-space interpolation followed by a lightweight post-merging alignment, ensuring robust cross-lingual transfer without the complexity of advanced merging mechanisms.
- **Comprehensive Evaluation:** We benchmark MLM on MMLU-ProX with LLaMA-3.2 1B and 3B models under multiple train/test splits, showing consistent gains over strong baselines in both Spanish and Hindi transfer.

2 Related Works

2.1 Multilingual LLM with Low Resource Languages

Recent advances in LLMs have demonstrated remarkable performance in high-resource languages such as English. However, their capabilities remain uneven across the multilingual spectrum, particularly for low-resource languages. This performance gap is well-documented by comprehensive multilingual benchmarks(Hu et al., 2020; Liang et al., 2020), including MMLU-ProX (Xuan et al., 2025), Global-MMLU (Singh et al., 2024), and BenchMAX (Huang et al., 2025), which evaluate LLMs on diverse tasks and languages. These benchmarks consistently reveal persistent disparities in accuracy and robustness for languages with limited training data, showing that despite improvements in model scale and architecture, LLMs still struggle to generalize effectively to low-resource languages. This highlights the ongoing need for more equitable and adaptable multilingual adaptation strategies. The challenge is further compounded by the scarcity and complexity of available corpora for many low-resource languages, making robust evaluation and adaptation both critical and technically demanding (Zhong et al., 2024; Li et al., 2024).

2.2 Model Merging

Recent advances in model merging have introduced several key techniques for combining task-specific or language-specific adapters. Task Arithmetic is a foundational approach where the difference between fine-tuned model weights and the base model weights is computed and then combined using a weighted sum, enabling the integration of multiple task vectors into a single model (Ilharco et al., 2022). Building on this, TIES (Yadav et al., 2023) refines the merging process by first trimming redundant parameters, resolving sign conflicts, and then aggregating only the aligned parameters, thereby reducing interference and improving robustness when merging multiple adapters. DARE (Yu et al., 2024) further enhances merging by randomly dropping a fraction of parameters and rescaling the remaining ones, which helps mitigate overfitting and parameter interference, making it a useful step before applying other merging methods.

More recently, research has shifted toward merging various LoRA adapters instead of entire models. Methods like LoRA Soups (Prabhakar et al., 2024) and LoRA-LEGO (Zhao et al., 2025) demonstrate how adapters trained on different tasks can be flexibly combined to improve generalization across tasks. Notably, CAT, as proposed in Prabhakar et al. (2024), learns optimal layer-wise weights for combining LoRA adapters, rather than simply refining existing merging techniques. However, most existing methods, including CAT, focus on general multitask settings and have not explicitly addressed the unique challenges of multilingual adaptation.

Our approach, MLM, distinguishes itself by decoupling task and language adaptation into separate LoRA adapters, specifically targeting robust multilingual performance through clear separation of task

and language learning. Unlike CAT, which optimizes merging coefficients, MLM directly fine-tunes the merged adapter itself, simplifying the optimization process with a small portion of the training set. Moreover, while advanced merging strategies such as CAT, TIES, and DARE could in principle be integrated into our framework, our study deliberately isolates the effects of initialization and adapter separation using simple parameter-space interpolation, leaving more sophisticated merging as a promising direction for future work.

3 Methodology

MLM is a modular fine-tuning framework that decouples task and language adaptation by training two separate LoRA-based adapters: a Task Adapter (TA) and a Language Adapter (LA). Rather than attempting to learn task semantics and linguistic variation simultaneously—which can obscure the source of errors and dilute training signals—MLM sequentially trains the TA with a high-resource language, then combines it with the model to train the LAs. This separation allows task knowledge learned in a high-resource language to be reused across different linguistic contexts, while the LA focuses solely on linguistic adaptation. To further refine the interaction between task and language knowledge, the final stage of training includes a lightweight alignment step using a small portion (15%) of bilingual training data. Figure 1 provides an overview of the training process.

3.1 Training Task Adapter (TA)

We first train the TA using a high-resource language such as English. The adapter is randomly initialized and optimized on a downstream task (e.g., multiple-choice QA or classification), while keeping the base LLM frozen. By leveraging the strong linguistic prior already present in the LLM due to its extensive pre-training on high-resource languages such as English, the TA can focus exclusively on capturing task-specific information. This ensures that the TA disentangles task logic from language variation, concentrating its capacity on the semantics of the downstream objective.

The outcome of this stage is a compact and transferable module that encapsulates task logic largely independent of language-specific variation. As a result, when adapting to other languages, including low-resource ones, it becomes sufficient to only train language-specific adapters, as the core task information is already preserved within the TA. This modularization ensures that task semantics are captured once in a reusable form, avoiding redundant retraining when new languages are introduced.

3.2 Training Language Adapter (LA)

Once the TA is trained, we freeze both its parameters and those of the base LLM. A new LA is then introduced and trained on the same task but using data from a low-resource language (e.g., Spanish or Hindi). At this stage, the TA and LA coexist as a merged adapter, with only the LA being updated. This guarantees that the task knowledge embedded in the TA is preserved while the LA adapts to the target language.

The role of LA. This setup allows the LA to function as a *linguistic bridge*. Because the TA already provides robust task semantics, the LA's role is to reinterpret inputs from the low-resource language into intermediate representations compatible with the frozen TA. In practice, the LA learns to adapt morphology, syntax, and word order variations into a form that the TA can immediately exploit for inference. By offloading task semantics to the TA, the LA can concentrate exclusively on linguistic alignment, thereby reducing redundancy between modules. This division of labor lowers sample complexity, improves data efficiency, and enables more effective transfer under conditions of data scarcity.

Notation and formalization. To formalize this process, let θ_{TA} and θ_{LA} denote the parameters of the TA and LA, respectively. Each LoRA adapter is parameterized by a pair of low-rank matrices (A, B), producing an effective update $\Delta W = BA$. We use $(A^{\text{TA}}, B^{\text{TA}})$ for the TA matrices and $(A^{\text{LA}}, B^{\text{LA}})$ for the LA matrices, while θ_{merged} represents the merged parameters obtained by interpolating θ_{TA} and θ_{LA}. This notation clarifies how the TA and LA are structurally compatible and how their contributions can be manipulated during training.

Initialization of the LA. To ensure structural compatibility, the TA and LA share the same LoRA architecture (e.g., rank and scaling). We investigate three initialization strategies that highlight different ways of exploiting the TA as a prior:

- **Zero Initialization.** $A^{\mathrm{LA}} = 0,\ B^{\mathrm{LA}} = 0$. At initialization, the merged adapter is identical to the TA, thereby preserving all task knowledge. During training, the LA gradually learns only the modifications required to capture the target language's properties. This initialization strongly biases the LA toward preserving TA semantics, ensuring that adaptation is minimal and targeted.
- **Orthogonal Initialization.** $A^{\mathrm{LA}} \perp A^{\mathrm{TA}},\ B^{\mathrm{LA}} = 0$. Here, the LA projection matrix is forced to be orthogonal to that of the TA. This encourages the LA to capture complementary information outside the TA's representational subspace, reducing redundancy and promoting diversity in what is learned. Conceptually, this initialization nudges the LA to explore language-specific factors that are distinct from task-related semantics.
- **Random Initialization.** $A^{\mathrm{LA}}, B^{\mathrm{LA}} \sim \mathcal{N}(0, \sigma^2)$. Serves as an uninformed baseline equivalent to conventional LoRA training, allowing us to contextualize the gains of informed initializations by contrasting them with a standard random setup.

In practice, we treat the choice of initialization as a hyperparameter in our experiments. This enables systematic comparison of strategies that preserve TA knowledge (zero), promote diversity (orthogonal), or rely on an uninformed baseline (random).

3.3 LoRA Merging Strategy

The TA and LA are merged directly in parameter space using a linear interpolation:

$$\theta_{\text{merged}} = \lambda_{\mathrm{TA}} \cdot \theta_{\mathrm{TA}} + \lambda_{\mathrm{LA}} \cdot \theta_{\mathrm{LA}}, \tag{1}$$

where λ_{TA} and λ_{LA} are scalar coefficients controlling the relative contributions of TA and LA. This formulation captures the simplest way to integrate task and language information into a single adapter.

While prior works often explore a wide range of interpolation coefficients $(\lambda_{\mathrm{TA}}, \lambda_{\mathrm{LA}})$, we restrict our study to two representative settings: $(0.5, 0.5)$ and $(1.0, 1.0)$. Empirically, the direct summation $(1.0, 1.0)$ consistently outperforms the normalized average $(0.5, 0.5)$. This indicates that preserving the full contributions of both adapters yields stronger generalization than balancing them down by normalization. The result suggests that task knowledge and linguistic alignment are complementary rather than competing signals, and that both should be preserved in their entirety.

Although more advanced merging strategies such as TIES Yadav et al. (2023) and DARE Yu et al. (2024) could be integrated into our framework, they introduce additional complexity and indirect updates to the LA. Our focus here is to isolate the effects of initialization and adapter separation, so we intentionally adopt simple merging. Nevertheless, we acknowledge that advanced merging approaches remain a promising direction for future work.

Post-merging alignment. After merging, we perform a lightweight alignment step inspired by CAT Prabhakar et al. (2024). We reuse a small portion of training data that contains both high-resource and low-resource examples, and jointly fine-tune the TA and LA for a single epoch. Unlike the main training phases, this alignment is not intended to relearn the task or the language. Instead, its goal is to refine the interface between the TA and LA by smoothing their interactions. The TA provides a stable reservoir of task knowledge, while the LA undergoes slight adjustments under bilingual supervision.

Without such an alignment step, the unilateral training of the LA followed by its direct merging with the frozen TA can lead to misaligned representations, causing partial loss of task information or suboptimal integration of linguistic features Pfeiffer et al. (2021); Xue et al. (2020). By briefly reusing mixed training data—that is, batches containing both high-resource and low-resource examples—post-merging alignment prevents this loss and ensures that both adapters contribute coherently.

It is worth noting that our approach differs from CAT in how alignment is performed: while CAT learns the merging coefficients between adapters, we instead directly fine-tune the merged adapter itself on a small portion of training data. This design choice simplifies the optimization procedure and provides a more explicit mechanism for aligning task and linguistic knowledge.

This additional step guarantees that the merged adapter achieves better integration of task and linguistic signals, thereby improving robustness across languages. Together, simple merging and lightweight alignment form a minimal yet effective recipe for multilingual transfer.

3.4 Design Motivation

The design of MLM is grounded in the principle of *structural disentanglement*: separating task semantics from language-specific processing to improve both transferability and scalability. Unlike joint fine-tuning approaches, which entangle adaptation objectives and often require updating the full model, MLM promotes modularity by isolating learning signals within lightweight adapters. This modular separation makes the training dynamics easier to interpret and control, while avoiding interference between task and language adaptation.

A key advantage of MLM is that a single TA trained on a high-resource language can be reused across multiple low-resource languages, thereby eliminating redundant retraining of task semantics. The cost of training the TA itself is comparable to training a standard adapter for a single language, but once obtained, it can be flexibly shared across languages without further duplication of effort. This not only improves computational efficiency but also clarifies the division of labor: the TA maintains stable task knowledge, while each LA is responsible solely for encoding language-specific variations.

The framework also supports extensibility. New LAs can be introduced for additional languages without the need to retrain existing components, and merging can be extended to multiple adapters in a plug-and-play fashion. This modularity makes MLM particularly suitable for multilingual systems that must scale to many languages under practical resource constraints.

Another important benefit is efficiency. Because task knowledge is already encapsulated in the TA and directly reused, learning for a new language only requires updating the LA. As shown in Section 4, this greatly accelerates adaptation: LA training required only a single epoch to converge, which is substantially faster than the training typically needed to learn merged adapters in prior LoRA-based methods. This demonstrates that MLM not only reduces computational overhead but also enables swift adaptation to new languages. In contrast, conventional merging approaches require each adapter to be trained independently before merging, leading to significantly greater training cost and slower cross-lingual transfer.

Efficiency and robustness via task-guided reuse. We view cross-lingual reasoning as reuse of task-specific computation learned once in a high-resource language and transferred to low-resource settings, freezing the TA and training only a lightweight LA so that adaptation reduces from task by language to language only and avoids relearning of task logic. This reuse makes adaptation efficient and raises reasoning performance in low-resource settings while attaining strong overall accuracy. The TA anchors task semantics, and the LA maps morphological and syntactic variation into the TA space, providing regularization when data are scarce. A lightweight post-merge alignment further stabilizes the TA–LA interface and improves cross-lingual consistency without costly additional tuning. Empirically, the design maintains or improves English performance while boosting target-language accuracy, yielding reliable transfer in low-resource regimes with minimal parameter updates. Next, we evaluate this design on multilingual reasoning benchmarks to quantify its efficiency, robustness under low-resource targets, and transfer quality.

4 Experiments and Results

4.1 Experiment Settings

We evaluate the proposed MLM approach on the MMLU-ProX benchmark under varying model sizes and train/test splits. Experiments use LLaMA-3.2 with two model sizes (1B and 3B) and two data-split regimes : 80:20 and 70:30 train/test splits. We report accuracy on both the English test set and the target-language test set, for each transfer scenario of Spanish [EN-ES] and Hindi [EN-HI].

All experiments were conducted on a server equipped with seven NVIDIA RTX A6000 GPUs, each with `48GiB` of VRAM. To identify the optimal configuration for each trainable model, we performed a hyperparameter search for the single-language LoRA adapters across all model sizes and data splits. We conducted a grid search over learning rates (`1e-4`, `3e-4`, `1e-5`, `3e-5`) and weight decay values (`0`, `1e-4`, `1e-6`), while keeping the batch size fixed at `32`. Each single-language adapter was

Table 1: LLaMA-3.2 1B model evaluated on MMLU-ProX datasets with 80:20 train:test split. The evaluation metric is *accuracy*, and bold numbers indicate the highest performance in each column.

Method	[EN-ES]		[EN-HI]	
	EN	ES	EN	HI
Zero Shot	0.11	0.11	0.11	0.11
Few Shot(3)	0.08	0.08	0.08	0.08
Few Shot(5)	0.07	0.07	0.07	0.07
Mixed Sample	0.33	0.27	0.30	0.21
Simple Sum	0.24	0.21	0.23	0.17
TIES	0.12	0.14	0.10	0.10
DARE	0.23	0.21	0.23	0.17
CAT	0.30	0.26	0.30	0.21
MLM	**0.34**	**0.29**	**0.33**	**0.22**

trained for 4 epochs, and the best-performing model was selected according to the lowest training loss observed at the final epoch. Consequently, the results reported in Tables 1, 2, and 3 represent the performance of these optimally selected models. The post-hoc merging baselines were constructed using the best single-language adapters identified through this process. For fair comparison, all LoRA-related hyperparameters were fixed across experiments: rank $r = 8$, scaling factor $\alpha = 16$, dropout probability 0.05, and target modules {`q_proj`, `k_proj`, `v_proj`, `up_proj`, `down_proj`}.

We compare MLM against diverse baseline methods. First, to measure the model's intrinsic capabilities without any parameter updates, we evaluate Zero-Shot and Few-Shot (3 and 5 shots) performance, where the latter provides in-context examples to guide the model's reasoning. As a standard fine-tuning approach, we include a Mixed Sample baseline, where a single LoRA adapter is trained on a combined bilingual dataset of English and the target language. We also evaluate several post-hoc merging techniques that combine independently trained adapters for each language without additional training. These include Simple Sum, which averages the adapter parameters; TIES, which reduces interference by resolving sign conflicts and trimming parameters; and DARE, which regularizes by dropping 30% of weights and rescaling the rest. Finally, we compare against CAT, a more advanced method that requires an additional training stage to learn the optimal layer-wise combination of two adapters, using a dedicated dataset sampled from 15% of each language's training data.

4.2 Results on the 1B Model

4.2.1 Main Results (80:20 split)

Table 1 reports the performance of each method using the LLaMA-3.2 1B model with an 80%/20% train-test split. We observe that MLM achieves the highest accuracy in both transfer scenarios.

Notably, even a direct addition of the two adapters (Simple Sum) yields a moderate transfer result, confirming that merging a task-specific and language-specific LoRA is a viable strategy. However, more advanced approaches demonstrate clear benefits. While CAT offers a notable improvement over this simple baseline, our MLM consistently widens the performance gap even further.

The Zero-Shot and Few-Shot baselines show weak performance, as they rely solely on the base model's limited cross-lingual capabilities without any fine-tuning. Furthermore, while TIES and DARE are more advanced merging algorithms than Simple Sum, they also fail to close the performance gap, suggesting their heuristics may not be optimal for the cross-lingual adaptation task.

While various baselines show mixed results, MLM consistently outperforms every baseline in both English and target-language accuracy. This superior performance underscores the benefits of its modular design, where the clear separation of task and language adaptation enables more effective and robust cross-lingual transfer.

Table 2: LLaMA-3.2 1B model evaluated on MMLU-ProX datasets with 70:30 train:test split. The evaluation metric is *accuracy*, and bold numbers indicate the highest performance in each column.

Method	[EN-ES]		[EN-HI]	
	EN	ES	EN	HI
Zero Shot	0.11	0.11	0.11	0.11
Few Shot(3)	0.07	0.07	0.07	0.08
Few Shot(5)	0.06	0.08	0.06	0.08
Mixed Sample	0.28	0.23	0.31	0.20
Simple Sum	0.24	0.21	0.23	0.15
TIES	0.12	0.13	0.10	0.10
DARE	0.24	0.21	0.23	0.15
CAT	0.31	0.26	0.30	0.21
MLM	**0.32**	**0.32**	**0.32**	**0.23**

4.2.2 Robustness to Data Scarcity (70:30 split)

To test robustness under even more low-resource conditions, we repeat the above experiments using 70% of the target data for training. Table 2 reports the performance of each method.

Our MLM method continues to outperform all alternatives, and the performance gap generally widens in this more challenging low-data scenario. This result highlights the robustness of MLM's modular architecture. By leveraging a strong, pre-trained Task Adapter, the framework enables highly sample-efficient learning for the Language Adapter, making it particularly effective when target-language data is scarce. Notably, the advantage of MLM is most pronounced under such limited-data conditions, where it achieves significantly higher accuracy than competing methods.

4.3 Results on the 3B Model

Finally, we also evaluated the scalability of our approach by applying it to the LLaMA-3.2 3B model (Table 3). We observe that scaling to the 3B model yields disproportionately larger gains in English compared to the target languages, thus widening the persistent cross-lingual performance gap for most baseline methods. Our MLM method continues to deliver strong cross-lingual transfer, achieving the highest accuracy across all methods for both Spanish [EN-ES] and Hindi [EN-HI] settings. Moreover, compared to baselines such as Mixed Sample, MLM remains markedly more sample-efficient, and this advantage becomes increasingly pronounced as the size of the base LLM scales up. This observation motivates a deeper examination of training efficiency and scalability, which we detail in the following section.

4.4 Training Efficiency and Scalability

Beyond accuracy, MLM's advantages in training efficiency are a direct consequence of its core design: the decoupling of task and language adaptation. Baselines like Mixed Sample face an entangled learning objective, where a single adapter must simultaneously master complex task semantics and bridge linguistic differences. This dual burden necessitates a prolonged fine-tuning process on a large bilingual dataset for each new language. CAT partially alleviates this by separating task and language adapters, but it still requires full, independent training for every new LA, often leading to redundant learning of task-aligned features, followed by an additional resource-intensive merging stage.

In contrast, MLM's efficiency stems from fundamentally simplifying the learning problem for each new language. By reusing a frozen, pre-trained Task Adapter that already encapsulates the necessary task logic, MLM offloads the most computationally expensive component. The Language Adapter's objective is thus narrowed to serving as a "linguistic bridge"—it only needs to learn the transformation that aligns the low-resource language with the TA's existing representation space. Because of this narrowed scope, training becomes a highly specialized and constrained optimization problem, enabling effective learning within a minimal fine-tuning budget (a brief adaptation phase followed by a swift alignment stage).

Table 3: LLaMA-3.2 3B model evaluated on MMLU-ProX datasets with 80:20 train:test split. The evaluation metric is *accuracy*, and bold numbers indicate the highest performance in each column.

Method	[EN-ES]		[EN-HI]	
	EN	ES	EN	HI
Zero Shot	0.14	0.14	0.14	0.12
Few Shot(3)	0.08	0.08	0.08	0.08
Few Shot(5)	0.07	0.07	0.07	0.07
Mixed Sample	**0.43**	**0.36**	**0.43**	0.28
Simple Sum	0.39	0.31	0.38	0.22
TIES	0.30	0.24	0.29	0.18
DARE	0.38	0.31	0.37	0.22
CAT	0.42	0.35	0.42	0.28
MLM	**0.43**	**0.36**	**0.43**	**0.29**

Finally, when comparing overall efficiency across methods, MLM demonstrates a clear advantage. Mixed Sample requires training on doubled data and restarting the process for each new language, while CAT demands full-epoch training of both task and language adapters, followed by additional fine-tuning. By contrast, MLM reuses a task adapter trained once for full epochs and only requires a single-epoch update for each new LA with a lightweight alignment stage, thereby ensuring greater sample efficiency as the number of target languages increases. .

5 Discussion and Future Work

Our results demonstrate that separating task and language adaptation provides both conceptual clarity and practical benefits for multilingual fine-tuning. By reusing a high-resource TA and constraining the LA to act as a linguistic bridge, MLM avoids redundant training and achieves robust transfer under data scarcity. Efficiency here is not only a matter of reduced computation but also a driver of generalization: minimal updates prevent overfitting and allow the LA to focus solely on linguistic alignment. This connection between efficiency and generalization is particularly important for low-resource languages, where over-parameterization often amplifies instability. Scaling experiments further show that while larger base models tend to widen cross-lingual gaps in baseline methods, MLM consistently narrows this disparity by anchoring transfer on the reusable TA. This suggests that modular strategies like MLM can complement scaling by ensuring that the benefits of larger models extend to all languages, not only high-resource ones. Moreover, the framework's plug-and-play design means that new languages can be supported with negligible additional cost, making it attractive for practical deployment in multilingual systems. Looking forward, integrating advanced merging techniques (e.g., TIES, DARE) and extending MLM to massively multilingual settings are promising directions, as is a deeper study of initialization strategies. These findings position MLM as both a strong baseline and a foundation for future modular approaches that balance scalability, efficiency, and equitable cross-lingual performance.

6 Conclusion

We present MLM, a modular framework that separates task and language adaptation into two lightweight adapters. By combining a reusable TA with a LA for target language through simple merging and a lightweight alignment stage, MLM enables efficient and robust transfer to low-resource languages without redundant retraining. Our experiments on MMLU-ProX demonstrate consistent gains across languages, model sizes, and training regimes, highlighting the framework's ability to deliver both scalability and efficiency. MLM shows that disentangling task and language knowledge offers a practical path toward more equitable multilingual LLMs, and provides a strong foundation on which future modular approaches can build.

7 Acknowledgement

This work was conducted during the collaboration between CARTE and LG Electronics, Toronto AI Lab. We would like to express our sincere gratitude to Manasa Bharadwaj for her invaluable supervision and guidance throughout the course of this project. This work was supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government(MSIT) (RS-2022-00143911, AI Excellence Global Innovative Leader Education Program)

References

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

Sanchit Ahuja, Divyanshu Aggarwal, Varun Gumma, Ishaan Watts, Ashutosh Sathe, Millicent Ochieng, Rishav Hada, Prachi Jain, Maxamed Axmed, Kalika Bali, et al. Megaverse: Benchmarking large language models across languages, modalities, models and tasks. *arXiv preprint arXiv:2311.07463*, 2023.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

Linzheng Chai, Jian Yang, Tao Sun, Hongcheng Guo, Jiaheng Liu, Bing Wang, Xinnian Liang, Jiaqi Bai, Tongliang Li, Qiyao Peng, et al. xcot: Cross-lingual instruction tuning for cross-lingual chain-of-thought reasoning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 23550–23558, 2025.

Adrian de Wynter, Ishaan Watts, Tua Wongsangaroonsri, Minghui Zhang, Noura Farra, Nektar Ege Altıntoprak, Lena Baur, Samantha Claudet, Pavel Gajdušek, Qilong Gu, et al. Rtp-lx: Can llms evaluate toxicity in multilingual scenarios? In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 27940–27950, 2025.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. *ICLR*, 1(2):3, 2022.

Junjie Hu, Sebastian Ruder, Aditya Siddhant, Graham Neubig, Orhan Firat, and Melvin Johnson. Xtreme: A massively multilingual multi-task benchmark for evaluating cross-lingual generalisation. In *International conference on machine learning*, pages 4411–4421. PMLR, 2020.

Xu Huang, Wenhao Zhu, Hanxu Hu, Conghui He, Lei Li, Shujian Huang, and Fei Yuan. Benchmax: A comprehensive multilingual evaluation suite for large language models. *arXiv preprint arXiv:2502.07346*, 2025.

Gabriel Ilharco, Marco Tulio Ribeiro, Mitchell Wortsman, Suchin Gururangan, Ludwig Schmidt, Hannaneh Hajishirzi, and Ali Farhadi. Editing models with task arithmetic. *arXiv preprint arXiv:2212.04089*, 2022.

Huiyuan Lai and Malvina Nissim. mcot: Multilingual instruction tuning for reasoning consistency in language models. *arXiv preprint arXiv:2406.02301*, 2024.

Zihao Li, Yucheng Shi, Zirui Liu, Fan Yang, Ninghao Liu, and Mengnan Du. Quantifying multilingual performance of large language models across languages. *arXiv e-prints*, pages arXiv–2404, 2024.

Yaobo Liang, Nan Duan, Yeyun Gong, Ning Wu, Fenfei Guo, Weizhen Qi, Ming Gong, Linjun Shou, Daxin Jiang, Guihong Cao, et al. Xglue: A new benchmark dataset for cross-lingual pre-training, understanding and generation. *arXiv preprint arXiv:2004.01401*, 2020.

Junhua Liu and Bin Fu. Responsible multilingual large language models: A survey of development, applications, and societal impact. *arXiv preprint arXiv:2410.17532*, 2024.

Niklas Muennighoff, Thomas Wang, Lintang Sutawika, Adam Roberts, Stella Biderman, Teven Le Scao, M Saiful Bari, Sheng Shen, Zheng-Xin Yong, Hailey Schoelkopf, et al. Crosslingual generalization through multitask finetuning. *arXiv preprint arXiv:2211.01786*, 2022.

Jonas Pfeiffer, Aishwarya Kamath, Andreas Rücklé, Kyunghyun Cho, and Iryna Gurevych. Adapterfusion: Non-destructive task composition for transfer learning, 2021.

Telmo Pires, Eva Schlinger, and Dan Garrette. How multilingual is multilingual bert? *arXiv preprint arXiv:1906.01502*, 2019.

Akshara Prabhakar, Yuanzhi Li, Karthik Narasimhan, Sham Kakade, Eran Malach, and Samy Jelassi. Lora soups: Merging loras for practical skill composition tasks. *arXiv preprint arXiv:2410.13025*, 2024.

Krithika Ramesh, Sunayana Sitaram, and Monojit Choudhury. Fairness in language models beyond english: Gaps and challenges. *arXiv preprint arXiv:2302.12578*, 2023.

Shivalika Singh, Angelika Romanou, Clémentine Fourrier, David I Adelani, Jian Gang Ngui, Daniel Vila-Suero, Peerat Limkonchotiwat, Kelly Marchisio, Wei Qi Leong, Yosephine Susanto, et al. Global mmlu: Understanding and addressing cultural and linguistic biases in multilingual evaluation. *arXiv preprint arXiv:2412.03304*, 2024.

Weihao Xuan, Rui Yang, Heli Qi, Qingcheng Zeng, Yunze Xiao, Yun Xing, Junjue Wang, Huitao Li, Xin Li, Kunyu Yu, et al. Mmlu-prox: A multilingual benchmark for advanced large language model evaluation. *arXiv preprint arXiv:2503.10497*, 2025.

Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. mt5: A massively multilingual pre-trained text-to-text transformer. *arXiv preprint arXiv:2010.11934*, 2020.

Prateek Yadav, Derek Tam, Leshem Choshen, Colin A Raffel, and Mohit Bansal. Ties-merging: Resolving interference when merging models. *Advances in Neural Information Processing Systems*, 36:7093–7115, 2023.

Le Yu, Bowen Yu, Haiyang Yu, Fei Huang, and Yongbin Li. Language models are super mario: Absorbing abilities from homologous models as a free lunch. In *Forty-first International Conference on Machine Learning*, 2024.

Yiran Zhao, Wenxuan Zhang, Huiming Wang, Kenji Kawaguchi, and Lidong Bing. Adamergex: Cross-lingual transfer with large language models via adaptive adapter merging. *arXiv preprint arXiv:2402.18913*, 2024.

Ziyu Zhao, Tao Shen, Didi Zhu, Zexi Li, Jing Su, Xuwu Wang, and Fei Wu. Merging loRAs like playing LEGO: Pushing the modularity of loRA to extremes through rank-wise clustering. In *The Thirteenth International Conference on Learning Representations*, 2025.

Tianyang Zhong, Zhenyuan Yang, Zhengliang Liu, Ruidong Zhang, Yiheng Liu, Haiyang Sun, Yi Pan, Yiwei Li, Yifan Zhou, Hanqi Jiang, et al. Opportunities and challenges of large language models for low-resource languages in humanities research. *arXiv preprint arXiv:2412.04497*, 2024.

Shaolin Zhu, Shaoyang Xu, Haoran Sun, Leiyu Pan, Menglong Cui, Jiangcun Du, Renren Jin, António Branco, Deyi Xiong, et al. Multilingual large language models: A systematic survey. *arXiv preprint arXiv:2411.11072*, 2024.