Mix Data or Merge Models? Optimizing for Performance and Safety in Multilingual Contexts

Anonymous EMNLP submission

Abstract

Large Language Models (LLMs) are increasingly used worldwide for diverse applications. However, ensuring their safe use continues to be a complex challenge. To tackle this, safety is often embedded into models as a "behavior" and is frequently overfit to harms prevalent in Western-centric datasets. In this work, we aim to address this by systematically exploring the potential of model merging in this diverse multi-task setting — considering safety in LLMs as a "task" and combining models trained for safety-specific tasks with those for more general-purpose tasks, all within a multilingual context. We categorize our experiments into two primary groups: objective-based and language-based, according to the fine-tuning objective of the models being merged. Our results demonstrate that objective-based merging is significantly more effective than data mixing, yielding improvements of up to 8% in general performance and 10% in safety. We also find that language-based merging is highly effective — by merging monolingual models, we achieve a 4% increase in general performance and 7% reduction in harm across all languages over the data mixing approach. Overall, our comprehensive study of model merging in the context of multilingual safety provides a useful framework for building strong and safe multilingual models without the need for retraining them.

1 Introduction

Large language models demonstrate strong multi-tasking capabilities across diverse domains (Brown et al., 2020; Radford et al., 2019). It is well established that equipping a model with any kind of capabilities with the standard paradigm of training requires copious amounts of data. Multi-tasking abilities typically arise from fine-tuning models on mixed datasets, which combine data from various sources and across many tasks (Raffel et al., 2023; Wang et al., 2019; Üstün et al., 2024). However, determining the optimal strategy for mixing datasets in multi-task training is often complex and resource-intensive, as it must ensure that all tasks benefit from the shared training process — especially in the context of safety, where the general performance of models often gets compromised in exchange for safety (Bai et al., 2022a; Tsipras et al., 2019; Bianchi et al., 2024; Ray and Bhalani, 2024; Üstün et al., 2024).

More recently, an emerging approach for enabling multi-tasking has focused on training distinct models for specific tasks and combining their parameters together using a predefined algorithm (Tam et al., 2023; Yang et al., 2024; Li et al., 2024a; Wan et al., 2024; Zhou et al., 2024; Davari and Belilovsky, 2024), to yield a resultant model that performs well on all of the considered tasks. This method has shown great promise in building models with new capabilities without incurring additional costs and challenges that accompany training from scratch. However, a key question remains – *how does it compare to traditional data mixing and weighting approaches?* We are, in particular, interested in exploring LLM safety with the perspective that *"safety"* can be conceptualized as an additional *"task-solving"* capability that a model can learn than a behavior that needs to be embedded via the method of model merging.

We evaluate the trade-offs between safety and general performance under severe multi-task constraints – optimizing for helpfulness and harmlessness in a *multilingual setting*. The inherent difficulties of handling multiple languages, each with its unique linguistic structures, cultural nuances, and potential biases, present a formidable task for aligning these models (Schwartz et al., 2022; Kotek et al., 2023; Khandelwal et al., 2023; Vashishtha et al., 2023; Khondaker et al., 2023; Üstün et al., 2024; Aryabumi et al., 2024; Singh et al., 2024). Mitigating harm across multiple languages is critical, given the wide adoption of LLMs across the world. However, a common issue in safety work

currently is the narrow focus on addressing it for English. And so, the challenges are compounded in this scenario by the scarce amount of safety data available across different languages (Singh et al., 2024). However, it is precisely because of these severe constraints that this presents an interesting setting to thoroughly evaluate the benefits of model merging.

We conduct an exhaustive study to compare traditional approaches for balancing multi-objective training by curating a wide set of training data mixtures with model merging methods for combining models trained on different subsets of data. Our large-scale evaluation runs across 6 languages from 5 different language families and encompasses both supervised fine-tuning and preference training across 4 different merging techniques. Through our comprehensive experimental setup, we summarize our key findings and contributions as follows:

1. Merging outperforms mixing. We find that model merging is more effective than weighting data mixtures for achieving a good balance between safety and generalizability in language models. The top-performing methods for individual objectives were TIES, which reduced harm by 10.4%, and Linear merging, which improved general performance by 8.6% over simple data mixing. The best approach for balancing both objectives was SLERP, which consistently achieved optimal trade-offs across different training strategies, with 3.1% reductions in harm and 7.0% gains in general performance over the data mixing approach.

2. Merging is effective at extending multilingual coverage. Instead of merging across objectives (safety-finetuned model and general-finetuned model), we experiment with merging across languages. Our findings indicate that when each model is trained on a mixture of safety and general data in a single language and then merged, it achieves improvements of up to 3.8% in general benchmarks and a reduction of up to 6.6% in harmful generations compared to a multilingually finetuned model.

3. Not all merging methods are equal. Some merging methods consistently result in net positive gains across both axes of performance (safety and general) simultaneously, while others display clear trade-offs. Model merging algorithms like Linear and TIES bring gains in only one dimension. For example, Linear merging resulting in improvements of up to 9% on general benchmarks but showing performance degradation as high as 8% on safety evaluations. Whereas merging models using DARE-TIES and SLERP is more effective in balancing the dual objectives, with SLERP showing the most significant improvements in both general performance and harm reduction (7% and 3.1% respectively). We see a similar pattern with linear merging.

2 *Mix* versus *Merge* Setup

In this section, we detail our experimental setup, which involves training models with various data mixtures targeting different objectives to establish the *"Mix"*, followed by merging some of these trained checkpoints into a single model to obtain the *"Merge"*. This setup serves as the foundation for our comprehensive comparison of merging methods' effectiveness in balancing safety and general performance in multilingual settings. Our experiments are set across both supervised fine-tuning (SFT) and offline preference tuning, specifically Direct Preference Optimization (DPO) (Rafailov et al., 2023).

2.1 Merging Approaches

We conduct extensive experiments with diverse data mixtures to create a pool of model candidates. From this pool, we merge the best-performing checkpoints using four different algorithms to produce the final merged models.

1) Linear Merge: Linear merging involves simple linear weighted averaging of model parameters, weighted by specified coefficients. This method is widely used in convex optimization and deep learning (Nagarajan and Kolter, 2021; von Oswald et al., 2022; Wortsman et al., 2022). This process is formulated as:

$$\theta_{\text{merged}} = \sum_{i=1}^{N} \alpha_i \theta_i \quad (1)$$

where α_i represents the weight assigned to the parameters of each model, with the constraint that $\sum_{i=1}^{N} \alpha_i = 1$. We conduct ablations by varying the values of α_i to investigate different weighting ratios for the base models.

2) Spherical Linear Interpolation (SLERP): This technique is used to smoothly blend two models by interpolating their weights along the shortest path on a high-dimensional sphere (White, 2016; Goddard et al., 2024). SLERP preserves

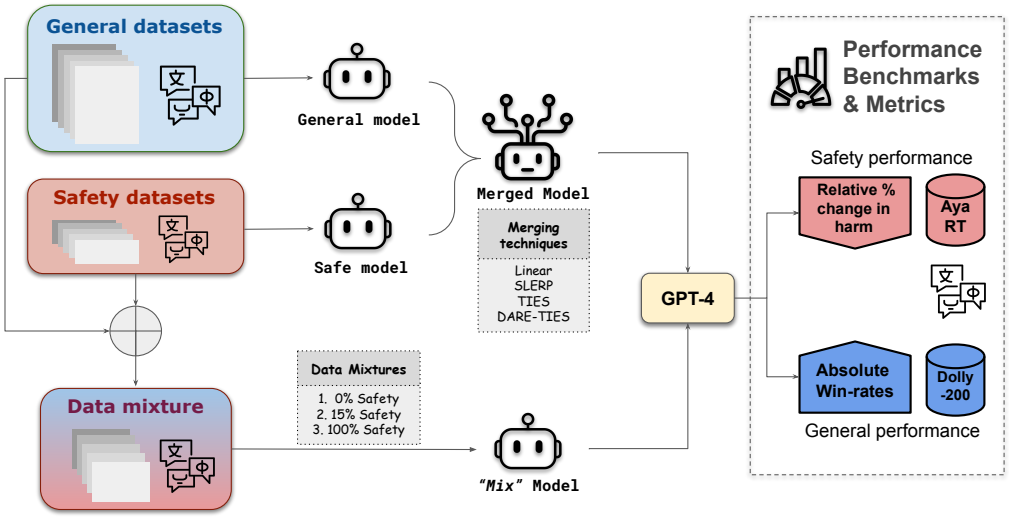

Figure 1: **Overview of our *Mix* versus *Merge* framework:** We analyze the differences in merging models on trained with specialized multilingual datasets, particularly in the context of safety, in contrast to those trained directly on mixtures of these datasets. We follow the LLM-as-a-judge approach for evaluating the performance of these models along two axes – general and safety.

each model's unique characteristics and geometric properties, even in complex spaces. The process involves normalizing the vectors to ensure equal length, calculating the angle Ω between them, and performing the interpolation as follows:

$$\theta_{\text{SLERP}}(t) = \frac{\sin((1-t)\Omega)}{\sin(\Omega)}\theta_1 + \frac{\sin(t\Omega)}{\sin(\Omega)}\theta_2 \quad (2)$$

SLERP typically merges only two models at a time. Here, $t \in [0, 1]$ determines the interpolation weight, with $t = 0$ using only *Model 1* and $t = 1$ using only *Model 2*. This method improves upon standard weight averaging by preserving the geometric integrity of the model.

3) TIES-Merging: This method efficiently combines multiple models by addressing parameter interference and sign conflicts, which occur when models suggest opposing adjustments to the same parameter due to task-specific fine-tuning (Yadav et al., 2023). The process begins by trimming parameters to retain only those with significant magnitude changes, i.e., the top-$k\%$. It then resolves sign conflicts by creating a consensus sign vector:

$$s = \text{sign}\left(\sum_{i=1}^{N} \text{sign}(\theta_i)\right) \quad (3)$$

Finally, it merges the parameters by averaging those that align with the consensus sign:

$$\theta_{\text{merged}} = s \cdot \frac{1}{N}\sum_{i=1}^{N} |\theta_i| \quad (4)$$

TIES-Merging ensures that only parameters contributing to the agreed-upon direction are included in the final model, enhancing performance.

4) DARE-TIES: This technique (Yu et al., 2024) builds upon TIES by applying dropout to the delta parameters before merging them using the TIES method. It reduces interference from redundant parameters and helps maintain the model's overall performance.

We apply gradient weighting to all merging methods except for Linear Merge. With weighting, we define a blend ratio to specify the merge between the model parameters. Gradient weighting dictates how that ratio changes across the specified values and uses linear interpolation to further establish a smoother gradient of blend ratios for merging the parameters. For example, if the blend ratio between *Model 1* and *Model 2* is defined as [0, 0.5, 1], this implies that the merge begins with 100% of *Model 2's* parameters, gradually transitioning to a 50-50 blend between the two and concluding with only *Model 1's* parameters at the end. For all merging methods, we conduct an exhaustive search over the set $\{0, 0.3, 0.5, 0.7, 1\}$ to determine the

Figure 2: *Mixing versus merging*: Safety and general performance of a *15% Safety Mix* model (§2.2) against SLERP merging, which emerges as the best method for balancing trade-offs, for both SFT and DPO based checkpoints. Lower is better for (a) and higher is better for (b). Both metrics are measured with respect to the Aya 23 base model.

optimal parameter contributions. Our experiments utilize the `mergekit` library from Arcee (Goddard et al., 2024).

2.2 Training Data

Safety dataset. We use the human-annotated prompts from the multilingual Aya Red-teaming dataset (Aakanksha et al., 2024) as seeds to synthetically generate pairs of adversarial prompts and contextually safe completions following the synthetic data generation pipeline outlined in Aakanksha et al. (2024).

General purpose dataset. Following previous works (Aakanksha et al., 2024), we use a sampled set of 10,000 English prompts from the *Ultrafeedback Binarized* (Cui et al., 2023; Tunstall et al., 2023) dataset translated into our target languages. This dataset will be referred to as the *"general-purpose"* dataset for the remainder of the paper.

Training data Mix. We study models trained on different mixtures of data - ***0% Safety Mix, 15% Safety Mix*** and ***100% Safety Mix***. The varying ratio of safety data simulates different objectives. For example, training with 100% safety data allows us to model an upper bound of expected harm mitigation and to obtain a model optimized for safety. In contrast, the 15% Safety mix consists of a combination of safety and general-purpose data in a 1:5 ratio – this represents a more real-world scenario typical of deployment settings and maintains a reasonable ratio for optimizing for both helpfulness and harmlessness of a model (Bai et al., 2022b). Unless specified otherwise, we use the 15% Safety mix as the baseline for our experimentation. The other mixes follow similar relationships between their naming and ratios.

2.3 Key Ablations

In order to study the relative merits of merging for different objectives across a wide set of languages, we conduct extensive ablations. We detail some of the most critical experiment variants below:

Objective-based merging. To evaluate the relative merits of merging on balancing dual-objectives, we merge models that have been separately optimized for general-purpose abilities and safety. This builds upon our multilingual 0% and 100% Safety Mixes (see Section 2.2) to balance the trade-offs between safety and general performance.

Language-based merging. Multilinguality remains one of the most challenging tasks in language modeling. We aim to determine whether language-specific models can be used off-the-shelf to incorporate language capabilities and explore how merging models based exclusively on different languages affects their downstream performance. Specifically, we investigate whether combining models optimized for both safety and general performance with a 15% language-specific safety mix for our target languages leads to better performance than training on a mixture of those languages. For clarity, to produce a multilingual model with safe and general-purpose abilities for English, French, and Spanish (referred to as the *EN-FR-SP* group later), we merge models optimized independently on a 15% Safety Mix for each of these languages.

2.4 Evaluation

Baseline. We evaluate the performance of all models against that of a previous checkpoint of

Type	Method	SFT		DPO	
		Aya RT (↓)	Dolly-200 (↑)	Aya RT (↓)	Dolly-200 (↑)
Training data mix	0% Safety	-41.4	70.0	-39.2	70.7
	15% Safety	-56.6	67.4	-54.69	71.0
	100% Safety	-64.4	64.8	-68.2	75.0
Merging	Linear	-49.1 (-7.5)	**76.0** (+8.6)	-48.6 (-6.1)	75.0 (+4.0)
	SLERP	**-58.2** (+1.2)	72.6 (+5.2)	-57.8 (+3.1)	78.0 (+7.0)
	TIES	-45.2 (-11.4)	74.9 (+7.5)	**-65.1** (+10.4)	63.6 (-7.4)
	DARE-TIES	-56.1 (-0.5)	70.0 (+2.6)	-55.9 (+1.2)	**78.5** (+7.5)

Table 1: Comparison of *Safety* and *General* performance across various methods. *Safety* performance is evaluated using the Aya Red-teaming (Aya RT) benchmark in terms of the "Relative Percentage Change in Harmful Generations" while *General* performance is evaluated with the Dolly-200 benchmark as "Absolute Win-rate Percentages". Both metrics are measured with respect to the Aya 23 base model. Scores are aggregated across six languages: English, Hindi, French, Spanish, Arabic, and Russian. Performance deltas, highlighted in color, represent differences from the 15% Safety Mix baseline.

the Aya 23 8B model (Aryabumi et al., 2024) – which henceforth acts as the baseline for all evaluations. This model is also treated as the pre-trained base model for all of our experiments. We note that this model was not optimized for safety. Hence, we measure the ability to minimize harmful model generations with respect to this model (% decrease).

We establish two axes of performance for our experiments — how *safe* model generations are and how well they perform on *general-purpose* benchmarks. We measure these with the following benchmarks:

Safety benchmark. We use the English prompts from the human-annotated *Aya Red-teaming dataset* (Aakanksha et al., 2024) and translate them into all of our target languages using the NLLB-3.3B model for an apples-to-apples comparison - i.e., for *Hindi, French, Spanish, Arabic* and *Russian*, resulting in a final set of 6 languages for evaluation. We measure the safety performance on this dataset as the negative relative percent change in harmful model generations with respect to the Aya 23 base model and report aggregated scores over all languages.

General benchmark. We use the *Multilingual Dolly-200 Eval* set (Singh et al., 2024; Üstün et al., 2024), which measures the open-ended generation capabilities of a language model. This dataset consists of a sample of 200 prompts from the Dolly-15k dataset translated into a number of languages, which then acts as a test bed for measuring the general performance of a language model. We use win-rates against the baseline to track performance changes.

To evaluate all experiments, we closely follow the evaluation framework of previous works (Aakanksha et al., 2024) and use the LLM-as-a-judge approach with GPT-4[1] as the evaluator. Given our dual axes of evaluation, safety and general performance, we instruct GPT-4 to classify model outputs as harmful or not to assess safety performance and to indicate an overall preference between two models' responses (experiment versus the Aya 23 base model) to measure the general performance.

3 Results and Discussion

In this section, we will present our results and discuss our findings.

3.1 Model merging wins over data mixing

Table 1 summarizes our findings and presents results for objective-based merging. The model trained on the 15% Safety Mix demonstrates strong performance on general tasks, achieving win rates of 67.4% for SFT and 71% for DPO. However, we see even greater improvements when merging checkpoints, with win-rates rising to 72.6% and 78%, respectively. We observe similar patterns in safety performance — the 15% Safety Mix model reduces harm by 56.6% for SFT and 54.7% for DPO. However, by merging checkpoints instead of mixing data, we achieve further reductions, reaching 58.2% for SFT and 57.8% for DPO. We evalu-

[1]https://platform.openai.com/docs/models/gpt-4-turbo-and-gpt-4

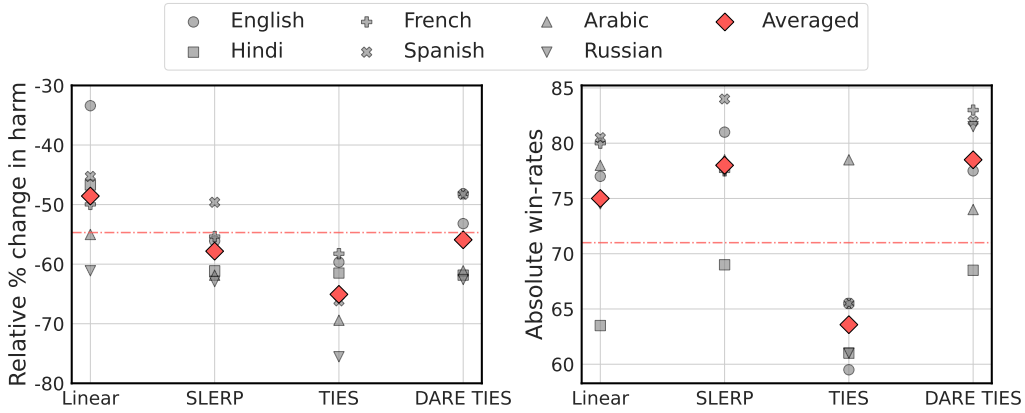

Figure 3: Comparison between different merging methods across safety and general performance with **DPO checkpoints**. Both metrics are measured with respect to the Aya 23 base model. Lower is better for the left and higher is better for the right. The red dashed line shows the model trained on a mix of safety and general data (*15% Safety Mix*).

ate the model with the best trade-off by considering the average percentage change of both objectives relative to the 15% Safety Mix model. Amongst the four methods evaluated, SLERP proved to be the most effective in balancing the two-fold objective of safety and general performance. Figure 2 shows the outcome of SLERP merging for both SFT and DPO checkpoints against the 15% Safety Mix baseline.

Overall, this supports the claim that merging models explicitly trained for different objectives outperforms building data mixtures aimed at the same goals. This is particularly compelling as a technique given previous studies have shown that optimizing for safety in a language model can negatively impact its general-purpose abilities (Bianchi et al., 2024; Ray and Bhalani, 2024; Bhardwaj et al., 2024; Üstün et al., 2024).

3.2 Not all merging methods are equal

Merging almost always benefits *general performance*, with all techniques but one (TIES) outperforming the 15% Safety Mix baseline (see Table 1). We observe gains as high as 7.5% in general performance when combining models with DARE-TIES, closely followed by SLERP with 7% gains. When focusing on *safety performance*, Table 1 illustrates that almost all merging methods perform superior to the 15% Safety Mix baseline, with the exception of Linear lagging behind by around 6%.

The dissimilarity of the checkpoints optimized for two different objectives can degrade performance when merging linearly, as the specialized parameter configurations for each task get diluted. On the other hand, we observe that TIES establishes substantial improvements in harm reduction by around 10% over the 15% Safety Mix. TIES strategically combines parameters based on their role in each task, preventing destructive interference while maintaining task-specific capabilities. When considering the trade-off between the two primary objectives — enhancing general performance and minimizing harm — SLERP emerges as the overall winner. This is mainly because SLERP finds intermediate points that balance both objectives' requirements by following the natural manifold of the parameter space rather than forcing direct averaging. The spherical interpolation in SLERP maintains relative distances between parameters, preventing one objective from dominating the other during merging.

3.3 Not all languages benefit equally

Next, we break down the multilingual evaluation and assess the effects of merging methods on individual languages. A detailed examination of Figure 3 (and Figure 6 in the Appendix) reveals that although overall improvements are consistent, the optimal trade-offs for different languages depend mostly on the underlying training regime of the model checkpoints used for merging.

Highest beneficiaries. For DPO, we find that *Russian* shows the most successful safety performance with a reduction of 15% over the 15% Safety Mix model with TIES merging. Spanish exhibits the most impressive improvements with around 6% with SLERP over the 15% Safety Mix baseline in general performance. For SFT, *Hindi* displays the

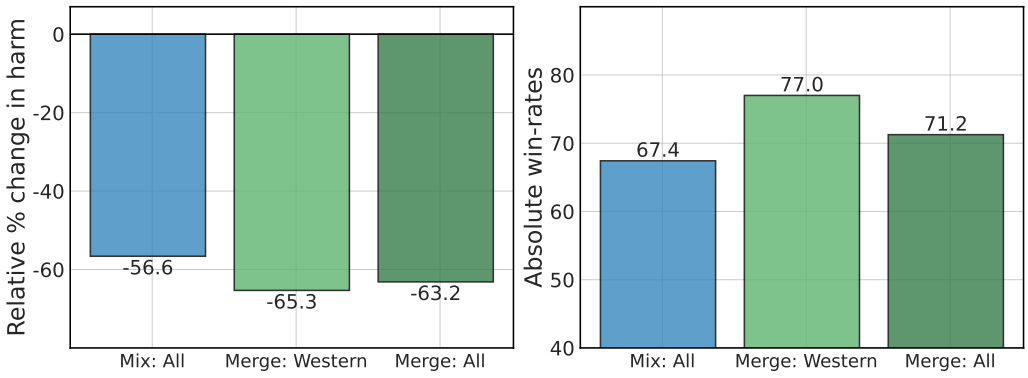

Figure 4: Monolingual model merging: We compare mixing vs merging with SFT checkpoints optimized for languages. The "[All]" bars represent model variants with all 6 languages – *English, Hindi, French, Spanish, Arabic* and *Russian*. "[EN,FR,SP]" is the pool of 3 "monolingual" models – *English, French* and *Spanish* . Both metrics are measured with respect to the Aya 23 base model. Lower is better for the left and higher is better for the right.

largest reduction in harm (12.14%) with SLERP over the 15% Safety Mix model. However, *Spanish* continues to reap the most benefits from merging with an improvement of 10% gains in general performance with both Linear and TIES.

Lowest beneficiaries. When merging DPO-based checkpoints (Figure 3), we surprisingly find *English* to benefit the least from merging across both axes of performance. We observe an overall decline of 24.87% in safety and 14.5% in general metrics compared to the 15% Safety Mix model with Linear and TIES merging respectively. For SFT checkpoints in the merging pool (Figure 6), we find that *Spanish* shows the lowest safety performance with TIES with an increase in harmful generations of around 16% while *Hindi* has the least gains in general performance with DARE-TIES with a decline of about 4% in comparison to the 15% Safety Mix.

It is worth noting that while merging leads to performance degradation in some languages compared to data mixing, it still delivers strong results, maintaining an absolute win-rate above 50% for all languages *relative to the base model*.

3.4 Merging monolingual models

Given the challenges posed by multilinguality and the linguistic and cultural variability introduced by each language, especially in the backdrop of safety, next we study the impact of merging models exclusively grounded in different languages on their downstream performance. For this set of experiments, we fine-tune our base model, Aya 23 8B, on monolingual data maintaining the 15% Safety Mix (§2.2) and use the resulting checkpoints for merging models across languages. For instance, to obtain a French-only model optimized for both safety and general performance, we fine-tune the model with only French samples, maintaining a 15% mix of safety in the training data. Extending this process for all languages yields 6 separately fine-tuned models on monolingual data.

Additionally, to understand the impact of scaling the number of languages during merging, we combine these models in gradation of two sets: one with 3 languages and another with 6. The 3-language set includes *English, French, Spanish* chosen for their closer familial ties and is referred to as the *"[EN,FR,SP]"* selection. The 6-language set comprises all our target languages — *English, French, Spanish, Hindi, Arabic* and *Russian* — and is termed *"[All]"* henceforth.

We focus on TIES for this set of experiments because its permutation-invariant nature helps us eliminate additional confounders and isolate the impact of language-based merging on overall performance. We use the same baseline as in previous experiments: a fine-tuned version of Aya 23 on a multilingual 15% Safety Mix. Figure 4 presents the results. We find that when compared to the base model, we successfully increase general performance and reduce harm generations across all variants. Merging 6 monolingual models (*"[All]"*) consistently outperforms the corresponding *"mix"* baseline, with safety metrics showing harm reductions as high as 6.6% and absolute improvements of 3.8% in general performance. However, we also observe some evidence of cross-lingual interference; merging 3 models (*"[EN,FR,SP]"*) yields better performance on both tasks compared to merging

6 models with differences of approximately 2% in safety and 6% in general performance. These results highlight model merging as an effective method for integrating a diverse set of languages without sacrificing performance on key metrics. The choice of languages and the number of models significantly influence the performance gains.

4 Related Work

Model Merging. Recent research has demonstrated success in developing innovative strategies to harness the collective power of multiple LLMs by suggesting methods for combining their unique strengths. This approach offers an efficient solution and has been widely explored for fine-tuned models sharing the same pre-trained base model, thereby sharing a part of their optimization trajectories (Frankle et al., 2020; Izmailov et al., 2019; Ilharco et al., 2023; Wortsman et al., 2022). Initial efforts focused on merging models with simple weighted averaging of the parameters (Wortsman et al., 2022; Matena and Raffel, 2022; Gupta et al., 2020) and showed dramatic performance gains for the resultant merged model. More recently, many works have investigated non-linear methods of merging models (White, 2016; Yadav et al., 2023; Yu et al., 2024) while aiming to improve general downstream performance. However, some recent works have focused on ensuring the safety of LLMs when merging, having demonstrated that misalignment transfers trivially from the base to the combined model in this process (Hammoud et al., 2024). Other works "realign" language models by fusing an initial aligned model with many task vectors based on the suitably identified safety subspace (Yi et al., 2024). Model merging has also been extended to a multilingual setting – for developing task-solving LLMs for low-resource languages without the availability of SFT data in the target languages (Tao et al., 2024). Our work distinguishes itself from prior approaches due to the complexity of the contrasting targets it seeks to satisfy — balancing safety and general-purpose objectives across a wide set of languages. To the best of our knowledge, no prior work has investigated the alignment of LLMs via model merging in a multilingual context while optimizing for a two-fold objective.

Multilingual Safety. With the increased pervasiveness of LLMs in recent times, the landscape of language model research has evolved with a heightened emphasis on safeguarding user experiences, thereby placing an increased focus on mitigating potential risks across diverse linguistic contexts. Several works (Deng et al., 2023; Liu et al., 2023) have investigated challenges around multilingual jailbreaks, and introduced novel frameworks and datasets for building robust mitigation strategies. Previous work has examined multilingual toxicity mitigation with a detailed comparison between SFT and retrieval-augmented-based methods (Pozzobon et al., 2024). It has been shown that LLMs tend to generate more harmful and irrelevant responses in low-resource languages when prompted maliciously (Shen et al., 2024). Techniques such as safety context distillation (Üstün et al., 2024) which harness synthetic data to institute safety guardrails into a model, have shown significant promise towards reducing the harmfulness in model generations. Overall, for a more standardized analysis of safety in multilingual settings, several benchmarks (Wang et al., 2023; Jain et al., 2024; Aakanksha et al., 2024) have been introduced and established in recent times. While methods such as SFT and DPO (Aakanksha et al., 2024; Li et al., 2024b) have been studied extensively for aligning language models, some recent works have also pivoted towards weight interpolation for the same objective and have demonstrated the effectiveness of adding a safety vector to compromised fine-tuned models for successful realignment (Bhardwaj et al., 2024). We direct our efforts towards the development of aligned language models by merging a diverse range of languages.

5 Conclusion

In this work, we demonstrated the effectiveness of model merging as a potential solution towards building highly-performant aligned language models across a wide range of languages. Through our comprehensive experimentation, we showed how models obtained as a result of merging exhibit superior performance on the dual axes of safety and general metrics. However, our experiments also revealed that there is variability in the trade-offs established by different merging algorithms, especially in a multilingual context. Additionally, we also demonstrated the success of combining models to extend language coverage while maintaining performance on the relevant metrics.

Limitations

While model merging offers a promising solution for better aligning LLMs, it poses a big challenge towards the interpretability of such models. The underlying weight distributions of neural networks are notoriously difficult to understand as they lack inherent meaning and merging only adds to the obscurity. Additionally, our work in its current shape does not include a hybrid set of experiments between the tasks and the languages, which would be an interesting setting to analyze the merits of merging in. Furthermore, it would also be valuable to study the impact of adding more tasks and/or objectives to the merging recipe on overall performance.

References

Aakanksha, Arash Ahmadian, Beyza Ermis, Seraphina Goldfarb-Tarrant, Julia Kreutzer, Marzieh Fadaee, and Sara Hooker. 2024. The multilingual alignment prism: Aligning global and local preferences to reduce harm.

Takuya Akiba, Makoto Shing, Yujin Tang, Qi Sun, and David Ha. 2024. Evolutionary optimization of model merging recipes.

Viraat Aryabumi, John Dang, Dwarak Talupuru, Saurabh Dash, David Cairuz, Hangyu Lin, Bharat Venkitesh, Madeline Smith, Jon Ander Campos, Yi Chern Tan, Kelly Marchisio, Max Bartolo, Sebastian Ruder, Acyr Locatelli, Julia Kreutzer, Nick Frosst, Aidan Gomez, Phil Blunsom, Marzieh Fadaee, Ahmet Üstün, and Sara Hooker. 2024. Aya 23: Open weight releases to further multilingual progress.

Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, et al. 2022a. Training a helpful and harmless assistant with reinforcement learning from human feedback. *arXiv preprint arXiv:2204.05862*.

Yuntao Bai, Saurav Kadavath, Sandipan Kundu, Amanda Askell, Jackson Kernion, Andy Jones, Anna Chen, Anna Goldie, Azalia Mirhoseini, Cameron McKinnon, Carol Chen, Catherine Olsson, Christopher Olah, Danny Hernandez, Dawn Drain, Deep Ganguli, Dustin Li, Eli Tran-Johnson, Ethan Perez, Jamie Kerr, Jared Mueller, Jeffrey Ladish, Joshua Landau, Kamal Ndousse, Kamile Lukosuite, Liane Lovitt, Michael Sellitto, Nelson Elhage, Nicholas Schiefer, Noemi Mercado, Nova DasSarma, Robert Lasenby, Robin Larson, Sam Ringer, Scott Johnston, Shauna Kravec, Sheer El Showk, Stanislav Fort, Tamera Lanham, Timothy Telleen-Lawton, Tom Conerly, Tom Henighan, Tristan Hume, Samuel R. Bowman, Zac Hatfield-Dodds, Ben Mann, Dario Amodei, Nicholas Joseph, Sam McCandlish, Tom Brown, and Jared Kaplan. 2022b. Constitutional ai: Harmlessness from ai feedback.

Rishabh Bhardwaj, Duc Anh Do, and Soujanya Poria. 2024. Language models are Homer simpson! safety re-alignment of fine-tuned language models through task arithmetic. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 14138–14149, Bangkok, Thailand. Association for Computational Linguistics.

Federico Bianchi, Mirac Suzgun, Giuseppe Attanasio, Paul Röttger, Dan Jurafsky, Tatsunori Hashimoto, and James Zou. 2024. Safety-tuned llamas: Lessons from improving the safety of large language models that follow instructions.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners.

Ganqu Cui, Lifan Yuan, Ning Ding, Guanming Yao, Wei Zhu, Yuan Ni, Guotong Xie, Zhiyuan Liu, and Maosong Sun. 2023. Ultrafeedback: Boosting language models with high-quality feedback.

MohammadReza Davari and Eugene Belilovsky. 2024. Model breadcrumbs: Scaling multi-task model merging with sparse masks.

Pala Tej Deep, Rishabh Bhardwaj, and Soujanya Poria. 2024. Della-merging: Reducing interference in model merging through magnitude-based sampling.

Yue Deng, Wenxuan Zhang, Sinno Jialin Pan, and Lidong Bing. 2023. Multilingual jailbreak challenges in large language models. *arXiv preprint arXiv:2310.06474*.

Jonathan Frankle, Gintare Karolina Dziugaite, Daniel M. Roy, and Michael Carbin. 2020. Linear mode connectivity and the lottery ticket hypothesis.

Charles Goddard, Shamane Siriwardhana, Malikeh Ehghaghi, Luke Meyers, Vlad Karpukhin, Brian Benedict, Mark McQuade, and Jacob Solawetz. 2024. Arcee's mergekit: A toolkit for merging large language models. *arXiv preprint arXiv:2403.13257*.

Vipul Gupta, Santiago Akle Serrano, and Dennis DeCoste. 2020. Stochastic weight averaging in parallel: Large-batch training that generalizes well.

Hasan Abed Al Kader Hammoud, Umberto Michieli, Fabio Pizzati, Philip Torr, Adel Bibi, Bernard Ghanem, and Mete Ozay. 2024. Model merging and safety alignment: One bad model spoils the bunch.

Gabriel Ilharco, Marco Tulio Ribeiro, Mitchell Wortsman, Suchin Gururangan, Ludwig Schmidt, Hannaneh Hajishirzi, and Ali Farhadi. 2023. Editing models with task arithmetic.

Pavel Izmailov, Dmitrii Podoprikhin, Timur Garipov, Dmitry Vetrov, and Andrew Gordon Wilson. 2019. Averaging weights leads to wider optima and better generalization.

Devansh Jain, Priyanshu Kumar, Samuel Gehman, Xuhui Zhou, Thomas Hartvigsen, and Maarten Sap. 2024. Polyglotoxicityprompts: Multilingual evaluation of neural toxic degeneration in large language models. *arXiv preprint arXiv:2405.09373*.

Dong-Hwan Jang, Sangdoo Yun, and Dongyoon Han. 2024. Model stock: All we need is just a few fine-tuned models.

Khyati Khandelwal, Manuel Tonneau, Andrew M. Bean, Hannah Rose Kirk, and Scott A. Hale. 2023. Casteist but not racist? quantifying disparities in large language model bias between india and the west.

Md Tawkat Islam Khondaker, Abdul Waheed, El Moatez Billah Nagoudi, and Muhammad Abdul-Mageed. 2023. Gptaraeval: A comprehensive evaluation of chatgpt on arabic nlp.

Hadas Kotek, Rikker Dockum, and David Sun. 2023. Gender bias and stereotypes in large language models. In *Proceedings of The ACM Collective Intelligence Conference*, CI '23. ACM.

Bingdong Li, Zixiang Di, Yanting Yang, Hong Qian, Peng Yang, Hao Hao, Ke Tang, and Aimin Zhou. 2024a. It's morphing time: Unleashing the potential of multiple llms via multi-objective optimization.

Xiaochen Li, Zheng-Xin Yong, and Stephen H. Bach. 2024b. Preference tuning for toxicity mitigation generalizes across languages.

Yi Liu, Gelei Deng, Zhengzi Xu, Yuekang Li, Yaowen Zheng, Ying Zhang, Lida Zhao, Tianwei Zhang, Kailong Wang, and Yang Liu. 2023. Jailbreaking chatgpt via prompt engineering: An empirical study. *arXiv preprint arXiv:2305.13860*.

Michael Matena and Colin Raffel. 2022. Merging models with fisher-weighted averaging.

Vaishnavh Nagarajan and J. Zico Kolter. 2021. Uniform convergence may be unable to explain generalization in deep learning.

Karl Pearson. 1900. X. on the criterion that a given system of deviations from the probable in the case of a correlated system of variables is such that it can be reasonably supposed to have arisen from random sampling. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 50(302):157–175.

Luiza Pozzobon, Patrick Lewis, Sara Hooker, and Beyza Ermis. 2024. From one to many: Expanding the scope of toxicity mitigation in language models.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D. Manning, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2023. Exploring the limits of transfer learning with a unified text-to-text transformer.

Ruchira Ray and Ruchi Bhalani. 2024. Mitigating exaggerated safety in large language models.

Reva Schwartz, Apostol Vassilev, Kristen K. Greene, Lori Perine, Andrew Burt, and Patrick Hall. 2022. Towards a standard for identifying and managing bias in artificial intelligence.

Lingfeng Shen, Weiting Tan, Sihao Chen, Yunmo Chen, Jingyu Zhang, Haoran Xu, Boyuan Zheng, Philipp Koehn, and Daniel Khashabi. 2024. The language barrier: Dissecting safety challenges of llms in multilingual contexts. *arXiv preprint arXiv:2401.13136*.

Shivalika Singh, Freddie Vargus, Daniel Dsouza, Börje F. Karlsson, Abinaya Mahendiran, Wei-Yin Ko, Herumb Shandilya, Jay Patel, Deividas Mataciunas, Laura OMahony, Mike Zhang, Ramith Hettiarachchi, Joseph Wilson, Marina Machado, Luisa Souza Moura, Dominik Krzemiński, Hakimeh Fadaei, Irem Ergün, Ifeoma Okoh, Aisha Alaagib, Oshan Mudannayake, Zaid Alyafeai, Vu Minh Chien, Sebastian Ruder, Surya Guthikonda, Emad A. Alghamdi, Sebastian Gehrmann, Niklas Muennighoff, Max Bartolo, Julia Kreutzer, Ahmet Üstün, Marzieh Fadaee, and Sara Hooker. 2024. Aya dataset: An open-access collection for multilingual instruction tuning.

Derek Tam, Mohit Bansal, and Colin Raffel. 2023. Merging by matching models in task parameter subspaces.

Mingxu Tao, Chen Zhang, Quzhe Huang, Tianyao Ma, Songfang Huang, Dongyan Zhao, and Yansong Feng. 2024. Unlocking the potential of model merging for low-resource languages.

Dimitris Tsipras, Shibani Santurkar, Logan Engstrom, Alexander Turner, and Aleksander Madry. 2019. Robustness may be at odds with accuracy.

Lewis Tunstall, Edward Beeching, Nathan Lambert, Nazneen Rajani, Kashif Rasul, Younes Belkada, Shengyi Huang, Leandro von Werra, Clémentine Fourrier, Nathan Habib, Nathan Sarrazin, Omar Sanseviero, Alexander M. Rush, and Thomas Wolf. 2023. Zephyr: Direct distillation of lm alignment.

Aniket Vashishtha, Kabir Ahuja, and Sunayana Sitaram. 2023. On evaluating and mitigating gender biases in multilingual settings.

Johannes von Oswald, Seijin Kobayashi, Alexander Meulemans, Christian Henning, Benjamin F. Grewe, and João Sacramento. 2022. Neural networks with late-phase weights.

Fanqi Wan, Xinting Huang, Deng Cai, Xiaojun Quan, Wei Bi, and Shuming Shi. 2024. Knowledge fusion of large language models.

Alex Wang, Jan Hula, Patrick Xia, Raghavendra Pappagari, R. Thomas McCoy, Roma Patel, Najoung Kim, Ian Tenney, Yinghui Huang, Katherin Yu, Shuning Jin, Berlin Chen, Benjamin Van Durme, Edouard Grave, Ellie Pavlick, and Samuel R. Bowman. 2019. Can you tell me how to get past sesame street? sentence-level pretraining beyond language modeling.

Wenxuan Wang, Zhaopeng Tu, Chang Chen, Youliang Yuan, Jen-tse Huang, Wenxiang Jiao, and Michael R Lyu. 2023. All languages matter: On the multilingual safety of large language models. *arXiv preprint arXiv:2310.00905*.

Tom White. 2016. Sampling generative networks.

Mitchell Wortsman, Gabriel Ilharco, Samir Yitzhak Gadre, Rebecca Roelofs, Raphael Gontijo-Lopes, Ari S. Morcos, Hongseok Namkoong, Ali Farhadi, Yair Carmon, Simon Kornblith, and Ludwig Schmidt. 2022. Model soups: averaging weights of multiple fine-tuned models improves accuracy without increasing inference time.

Shitao Xiao, Zheng Liu, Peitian Zhang, and Xingrun Xing. 2023. Lm-cocktail: Resilient tuning of language models via model merging.

Prateek Yadav, Derek Tam, Leshem Choshen, Colin Raffel, and Mohit Bansal. 2023. Ties-merging: Resolving interference when merging models.

Enneng Yang, Zhenyi Wang, Li Shen, Shiwei Liu, Guibing Guo, Xingwei Wang, and Dacheng Tao. 2024. Adamerging: Adaptive model merging for multi-task learning.

Xin Yi, Shunfan Zheng, Linlin Wang, Xiaoling Wang, and Liang He. 2024. A safety realignment framework via subspace-oriented model fusion for large language models.

Le Yu, Bowen Yu, Haiyang Yu, Fei Huang, and Yongbin Li. 2024. Language models are super mario: Absorbing abilities from homologous models as a free lunch.

Yuyan Zhou, Liang Song, Bingning Wang, and Weipeng Chen. 2024. Metagpt: Merging large language models using model exclusive task arithmetic.

Ahmet Üstün, Viraat Aryabumi, Zheng-Xin Yong, Wei-Yin Ko, Daniel D'souza, Gbemileke Onilude, Neel Bhandari, Shivalika Singh, Hui-Lee Ooi, Amr Kayid, Freddie Vargus, Phil Blunsom, Shayne Longpre, Niklas Muennighoff, Marzieh Fadaee, Julia Kreutzer, and Sara Hooker. 2024. Aya model: An instruction finetuned open-access multilingual language model.

A Additional Ablations

A.1 Comparison of merging applied to DPO and SFT.

Model merging is a highly adaptable technique that can be applied at any stage of the training process owing to its simple input requirement of model checkpoints. To determine the optimal stage for maximizing its benefits, we merge and evaluate SFT and DPO checkpoints independently as these techniques have shown great success towards the alignment of language models (Aakanksha et al., 2024; Shen et al., 2024).

A.2 Sensitivity to hyperparameters.

Previous works (Ilharco et al., 2023) have shown that merging is sensitive to the hyperparameters involved and have developed sophisticated algorithms (Akiba et al., 2024; Xiao et al., 2023; Davari and Belilovsky, 2024) to find the optimal values for the same. To this end, we seek to find the impact of varying the weighting scheme of Linear merging on both general performance and safety.

A.3 Comparison between additional merging methods

In the main text, we focus on standard model merging methods like weight averaging, SLERP etc. and explore the potential of merging through them. However, we extend our study here to measure the impact of additional merging methods in order to debias our findings from a limited subset of merging methods.

B Additional Results

B.1 DPO merges are more robust than SFT merges

Given the versatility of merging, which can be applied to any grouping of checkpoints, we separately compare merging gains when applied to models optimized with SFT and DPO (Table 1). We find that DPO merging better preserves safety constraints while improving performance, while SFT merging shows a performance-safety tradeoff. This suggests that DPO training creates more stable and consistent parameter spaces for merging than SFT.

More concretely, our experiments show larger consistent improvements when merging DPO checkpoints, with average gains of 2.8% and 2.2% over the base model across the four merging methods assessed for general performance and safety, respectively. While merging SFT checkpoints also resulted in significant general performance gains, averaging around 6%, it led to an average increase of 4.6% in harmful generations relative to the 15% Safety Mix model.

B.2 Impact of safety model weight on merging

Here, we evaluate how model coefficients during merging impact our "objective-based" merging approach on our dual axes of performance. Figure 5 illustrates that the safety performance of the merged model is greatly enhanced when a higher weight is attributed to the safety model. The merged model can mitigate harm more effectively than the 15% Safety Mix baseline, even with a normalized weighting for the constituent safety model as low as 0.3. For general performance, we observe that increasing the weight of the safety-focused model leads to a decrease in the model's performance on general tasks. However, across all weightings, merging models consistently outperforms the data mix run.

B.3 SLERP establishes the best trade-offs

In this section, we experimented with some additional merging methods with the DPO checkpoints. Results can be found in Table 2. Here, Task Arithmetic (Ilharco et al., 2023) seems to perform the best, quite similarly to SLERP (but inferior), while the other two seem lacking. Model Stock's (Jang et al., 2024) performance is contingent on the algorithm exploiting certain geometric properties of the weight space to find the optimal set of weights with which to combine the models. However, this approach might overlook nuanced interactions between model parameters, potentially limiting performance gains in complex scenarios. In case of DELLA-merging (Deep et al., 2024), it relies heavily on magnitude based pruning under the assumption that magnitudes correlate to importance, which may not necessarily be true.

B.4 Continual training after merging

In this section, we examine the dynamics of merging and preference training, focusing on the best ways to integrate both into the training pipeline. More specifically, we use DPO to assess whether continual preference tuning of a merged checkpoint results in stronger models compared to a merged model where the constituent models were individually preference-tuned. As can be seen in Table 3, our experiments demonstrate that continually

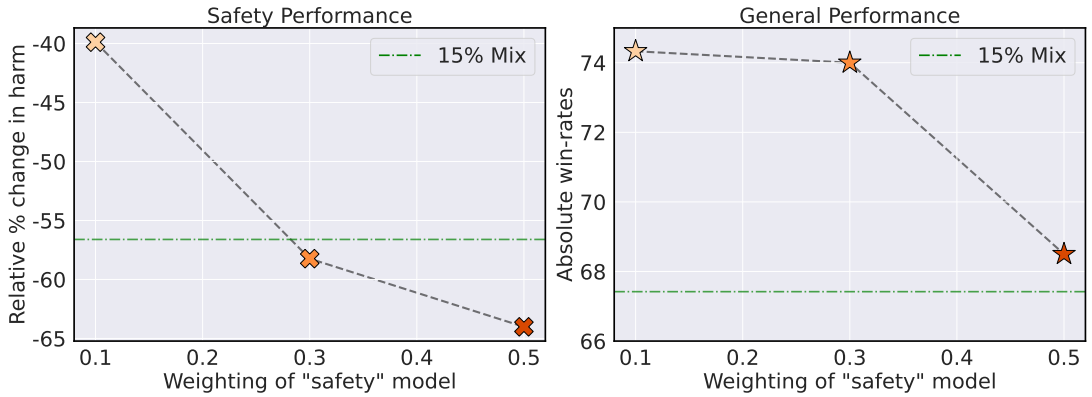

Figure 5: Ablation: Effect of *"safety weighting"* while Linear merging. We vary the weight assigned to the *100% Safety* model while merging linearly and measure the impact of the same. Both metrics are measured with respect to the Aya 23 base model. Lower is better for the left and higher is better for the right.

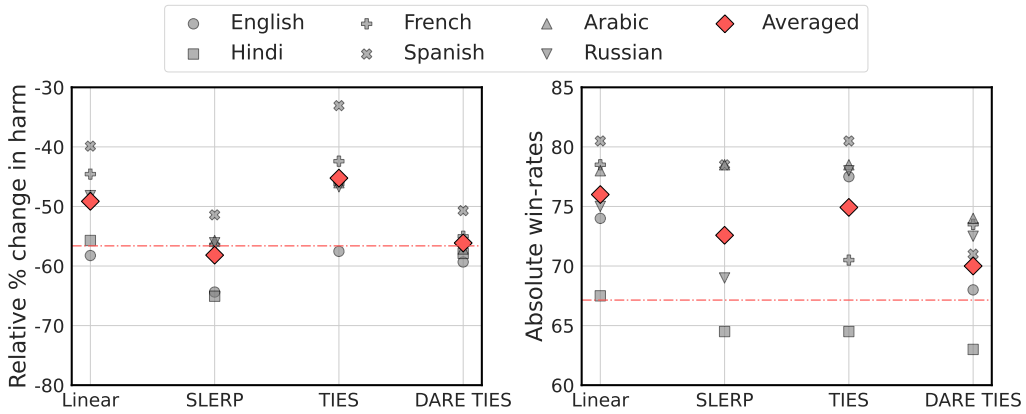

Figure 6: Comparison between different merging methods across safety and general performance with **SFT checkpoints**. Both metrics are measured with respect to the Aya 23 base model. Lower is better for the left and higher is better for the right. The red dashed line represents the model trained on a mixture of safety and general data (*15% Safety Mix*).

Type	**Method**	**DPO**	
		Aya RT (↓)	**Dolly-200 (↑)**
Merging	Task Arithmetic	-53.3 (-1.3)	**78.8** (+7.8)
	Model Stock	**-4.2** (-50.5)	45.2 (-25.8)
	DELLA	-19.7 (-35.0)	73.0 (+2.0)

Table 2: Comparison of *Safety* and *General* performance across some additional merging methods on DPO checkpoints. *Safety* performance is evaluated using the Aya Red-teaming (Aya RT) benchmark in terms of the "Relative Percentage Change in Harmful Generations" while *General* performance is evaluated with the Dolly-200 benchmark as "Absolute Win-rate Percentages". Both metrics are measured with respect to the Aya 23 base model. Scores are aggregated across six languages: English, Hindi, French, Spanish, Arabic, and Russian. Performance deltas, highlighted in color, represent differences from the 15% Safety Mix baseline (refer Table 1).

preference-tuning the models *after* performing the merge yields better outcomes in terms of alignment. The *"after"* merging variant (SFT → ⟨merge⟩ → DPO) shows better safety performance by reducing harmful generations by 6.5% whereas the *"before"* merging variant (SFT → DPO → ⟨merge⟩) exhibits a 3.1% decrease. We observe improvements in the general performance of both variants, with the *"after"* merge variant yielding a 3% increase, and the *"before"* merge variant achieving a 7% increase.

Training pipeline	Aya RT (↓)	Dolly-200 (↑)
SFT → ⟨merge⟩	-58.2 (+1.6)	72.6 (+5.2)
SFT → DPO → ⟨merge⟩	-57.8 (+3.1)	**78.0 (+7.0)**
SFT → ⟨merge⟩ → DPO	**-61.2 (+6.5)**	74.0 (+3.0)

Table 3: Comparison between offline preference tuning models before (row 2) and after (row 3) merging. The scores represent absolute "% relative change in harm" with respect to the Aya 23 base model while the gains in parentheses are reported with respect to the 15% Safety Mix model. The merging technique used here is SLERP.

B.5 Language-based breakdown of "objective-based" merging

Tables 4 - 7 show the language-based breakdown of our "objective-based" merging method.

C Computational Comparison

We would like to highlight here that there is little to no additional computational cost associated with model merging given the input models are readily available.

For context, when using GPUs (let's say 80GB A100s), the upper bound on merging 8B models is 180s or 3 minutes (under a minute on average), requiring only a single GPU. However, the lower bound on supervised fine-tuning the same 8B models across 8 such GPUs is 30 minutes and when preference tuning (with DPO) is at least 16 hours. For the most part, it would be trivial enough to carry out merging on CPUs without significant changes in time taken. And so, it would be much easier and cheaper to search through the "merging" space than to train (SFT or DPO, let alone SFT + DPO) even a second "version B" model. For every model that you would consider "training" with SFT or DPO, there could undoubtedly be a dozen more or an impressively large hard-to-count number with merging respectively.

Overall, merging models is a relatively inexpensive operation if the models are at hand. We also note that there is no extra cost associated with merging at inference time in terms of memory or compute.

D Statistical Significance Testing

We performed extensive significance testing for all the findings that we present in our paper. Specifically, we performed a pairwise Chi-squared test (Pearson, 1900) between X and Y (which we define below) with α = 0.05 across all languages separately, since our prediction variables for both metrics were categorical – [harmful / not harmful] for safety performance and [win, loss, tie] for general performance. This implies that all results with p-values less than 0.05 here are statistically significant. To be explicit, the null hypothesis states that there is no significant difference between the observed variables X and Y.

For Figure 2, we ran two separate tests with X = *"15% Mix"* model and Y equal to one of "SLERP - SFT" and "SLERP - DPO" separately, for comparing the *"Mix"* and the "objective-based" *"Merge"* variant. All cases rejected the null hypothesis with p-values ranging from $5e{-3}$ to $1e{-52}$ (across both safety and general performance as well as different languages), indicating statistically significant results.

For results in Figures 3 and 6, we performed separate tests for the SFT and DPO checkpoints with X = "SLERP" and Y = "TIES", to compare the significance of the results between the most effective and least effective merging methods. All cases again rejected the null hypothesis with p-values ranging from $1e{-3}$ to $4e{-51}$ across safety and general performance for all languages.

For Figure 4, we again ran separate tests with [X = "Mix: All" and Y = "Merge: Western"] for [EN, FR, SP] and another with [X = "Mix: All" and Y = "Merge: All"] with all 6 languages for comparing the performance between the *"Mix"* and the "language-based" *"Merge"* variants. All cases across both tests rejected the null hypothesis with p-values ranging from $1e{-4}$ to $3e{-49}$.

Type	Method	English	Hindi	Arabic	French	Spanish	Russian
Training data mix	0% Safety	-58.5	-46.8	-41.4	-33.3	-32.3	-34.0
	15% Safety	-69.1	-47.3	-57.2	-51.4	-53.5	-58.1
	100% Safety	-72.7	-51.4	-59.8	-55.7	-70.7	-72.7
Merging	Linear	-58.2	-55.7	-48.2	-44.6	-39.9	-48.2
	SLERP	-64.4	-65.1	-55.7	-56.4	-51.4	-56.1
	TIES	-57.5	-45.7	-46.0	-42.4	-33.1	-46.7
	DARE-TIES	-59.3	-57.9	-57.2	-55.0	-50.7	-56.8

Table 4: Comparison of *safety* performance with "objective-based merging" across various methods on the Aya Red-teaming benchmark in terms of the "Relative Percentage Change in Harmful Generations" with respect to the Aya 23 base model at a language level. All methods utilize SFT checkpoints.

Type	Method	English	Hindi	Arabic	French	Spanish	Russian
Training data mix	0% Safety	68.5	57.5	76.5	73.0	77.0	67.5
	15% Safety	69.5	67.0	69.0	68.5	68.5	62.0
	100% Safety	66.5	56.0	62.5	72.0	66.0	66.0
Merging	Linear	74.0	67.5	78.0	78.5	80.5	75.0
	SLERP	72.5	64.5	78.5	72.5	78.5	69.0
	TIES	77.5	64.5	78.5	70.5	80.5	78.0
	DARE-TIES	68.0	63.0	74.0	73.5	71.0	72.5

Table 5: Comparison of *general* performance with "objective-based merging" across various methods on the Multilingual Dolly-200 in terms of "Absolute Win-rates" against the Aya 23 base model at a language level. All values are represent percentages. All methods utilize SFT checkpoints.

Type	Method	English	Hindi	Arabic	French	Spanish	Russian
Training data mix	0% Safety	-59.1	-45.6	-36.5	-28.7	-28.6	-34.4
	15% Safety	-68.8	-42.7	-57.9	-42.2	-54.9	-58.1
	100% Safety	-76.4	-62.8	-61.3	-62.4	-67.0	-77.9
Merging	Linear	-33.4	-46.7	-55.0	-50.0	-45.3	-61.1
	SLERP	-56.1	-61.1	-61.8	-55.4	-49.6	-62.9
	TIES	-59.7	-61.5	-69.4	-58.2	-66.2	-75.5
	DARE-TIES	-53.2	-61.8	-61.1	-48.2	-48.3	-62.6

Table 6: Comparison of *safety* performance with "objective-based merging" across various methods on the Aya Red-teaming benchmark in terms of the "Relative Percentage Change in Harmful Generations" with respect to the Aya 23 base model at a language level. All methods utilize DPO checkpoints.

Type	Method	English	Hindi	Arabic	French	Spanish	Russian
Training data mix	0% Safety	71.5	56.0	72.0	75.0	79.5	70
	15% Safety	74.0	61.0	71.5	73.0	78	68.5
	100% Safety	77.0	68.0	77.5	72.0	79.5	77
Merging	Linear	77.0	63.5	78.0	80.0	80.5	74.5
	SLERP	81.0	69.0	79.5	77.5	84	77.5
	TIES	59.5	61.0	69.0	65.6	65.5	61.0
	DARE-TIES	77.5	68.5	78.5	83.0	82	81.5

Table 7: Comparison of *general* performance with "objective-based merging" across various methods on the Multilingual Dolly-200 in terms of "Absolute Win-rates" against the Aya 23 base model at a language level. All values are represent percentages. All methods utilize DPO checkpoints.