

Cross-Lingual Knowledge Transfer for Arabic Summarization via Model Merging and Portable Reward Tuning

Ahmad Almoustafa¹ Takuya Matsuzaki¹

¹Tokyo University of Science

1425501@ed.tus.ac.jp matuzaki@rs.tus.ac.jp

Abstract

Arabic summarization remains challenging due to limited supervised data and the cost of fine-tuning large language models (LLMs). We study model merging as a way to transfer English summarization behavior to Arabic without retraining large models. Starting from a multilingual decoder-only LLM, we train two parameter-efficient summarization adapters: one on Arabic news and one on English news, and apply a sparse, conflict-aware merge that selectively combines adapter updates in targeted attention components. On an Arabic news summarization test set, the merged adapter outperforms an Arabic-only adapter on ROUGE, BLEU, BERTScore, and LLM-as-a-judge protocol. To scale within the same model family, we also merge Arabic and English reward adapters and use the merged reward to steer a larger frozen model under Portable Reward Tuning (PRT) at inference time, yielding consistent gains over using an Arabic-only reward.

1 Introduction

Arabic is spoken by hundreds of millions of people and is central to news and public discourse across more than twenty countries. Yet, compared to English and other high-resource languages, Arabic remains underrepresented in large-scale resources and benchmarks for generative tasks such as summarization [1]. This gap matters in practice, where readers and organizations often need concise, faithful summaries of long, information-dense texts.

Progress in Arabic summarization has benefited from Arabic-focused pretraining and domain adaptation of language models, including AraBERT [2], AraBART [3], and FinAraT5 [4]. Yet two challenges persist: most systems rely mainly on Arabic supervision despite abundant English summarization data and strong conventions [5], and

scaling multilingual LLMs makes full fine-tuning costly. Parameter-efficient methods such as Low-Rank Adaptation (LoRA) [6] reduce cost via small adapters on frozen backbones, but still produce language-specific modules with no principled way to compose cross-lingual knowledge.

Model merging offers a complementary alternative: it combines independently trained updates directly in parameter space, often without additional gradient-based tuning [7]. In particular, TIES-Merging (TrIm, Elect Sign-merge) [8] provides a sparse, conflict-aware strategy for composing updates, making it a natural candidate for merging LoRA adapters trained on different languages. Despite promising results in other settings, cross-lingual merging has not been systematically studied for Arabic summarization as a mechanism for transferring English summarization behavior into Arabic.

We study cross-lingual model merging for Arabic summarization using Aya Expanse 8B [9] as a decoder-only backbone. We train Arabic and English LoRA summarization adapters and merge them with a head-wise, projection-aware TIES procedure controlled by a mixing coefficient λ . To scale without fine-tuning a larger generator, we instantiate Portable Reward Tuning (PRT) [10] with Arabic and English LoRA reward adapters trained on Aya Expanse 8B, merge them in the same way, and use the merged reward to steer a frozen Aya Expanse 32B model at inference time.

Our contributions are:

- Cross-lingual, head-wise, projection-aware merging of Arabic and English LoRA summarization adapters, which is shown to be beneficial on Aya Expanse 8B.
- Reward-level merging that further enhances performance by steering a larger frozen generator via PRT-style decoding.
- Controlled ablations and evaluation that characterize when cross-lingual merging is most beneficial.

2 Preliminaries

We review model merging and Portable Reward Tuning (PRT), which we use as building blocks.

2.1 Model Merging

Model merging composes independently adapted models that share a backbone by operating directly in parameter space [7]. Given base parameters θ_0 and adapted variants θ_A and θ_B , their task vectors $v_A = \theta_A - \theta_0$ and $v_B = \theta_B - \theta_0$ can be combined as

$$\theta_{\text{merge}} = \theta_0 + \alpha v_A + \beta v_B,$$

where α and β control the contribution of each update. Naïve linear merging can degrade performance when updates conflict. We therefore use conflict-aware merging via TIES-Merging [8], which trims each task vector to retain only high-magnitude updates, elects a consensus sign per parameter, and merges only updates consistent with that sign while discarding conflicting values. We also consider DARE-TIES [11], which applies DARE (Drop And Rescale) before TIES by randomly dropping a fraction of task vector components and rescaling the remainder, improving robustness under stronger sparsification.

2.2 Portable Reward Tuning

Portable Reward Tuning (PRT) [10] concentrates task-specific learning into an explicit reward model while keeping the generator frozen. Given a pretrained predictor $\pi_{\text{pt}}(y | x)$ and reward $r_\theta(x, y)$, PRT defines the induced predictor as the closed-form KL-regularized maximizer

$$\pi_\theta(y | x) \propto \pi_{\text{pt}}(y | x) \exp(\lambda^{-1} r_\theta(x, y)),$$

equivalently adding scaled reward logits to the base logits before the softmax. The reward is trained so that the induced predictor minimizes the same supervised loss as standard fine-tuning, enabling the learned reward to be paired with a (possibly larger) foundation model at inference time.

3 Method

We present our two-stage approach: (i) cross-lingual merging of Arabic and English LoRA summarization adapters on Aya Expanse 8B, and (ii) cross-lingual merging of reward adapters to steer a frozen Aya Expanse 32B generator at inference time.

3.1 Training Adapters

On the frozen Aya-8B backbone, we train two LoRA summarization adapters: Arabic on ARASUM and English on CNN/DailyMail, using identical LoRA and optimization settings. Hence, the adapters differ only in supervision. We also train LoRA reward adapters on Aya-8B for Arabic (ARASUM) and English (CNN/DailyMail) using PRT-style cross-entropy on fused logits as the loss function.

3.2 Cross-Lingual LoRA Adapter Merging

Let $\Delta W_m^{(\text{AR})}$ and $\Delta W_m^{(\text{EN})}$ denote the effective LoRA-induced weight updates for a linear module m . We merge updates only within self-attention projections, optionally restricted to selected attention heads. We consider two merge granularities: single-projection merging (one of $\{q, k, v, o\}$ projections) and multi-projection merging (merging a specified subset of projections).

For each selected parameter block, we apply TIES-style conflict-aware merging by: (i) trimming each update to retain only its top-magnitude entries (trim rate ρ), and (ii) electing a sign and discarding sign-inconsistent updates (resolving conflicts by keeping the larger-magnitude value). This yields a conflict-aware merged update $\Delta W_m^{(\text{TIES})}$. We then interpolate between the Arabic update and the merged update:

$$\Delta W_m^{(\text{merge})} = (1 - \lambda) \Delta W_m^{(\text{AR})} + \lambda \Delta W_m^{(\text{TIES})}.$$

The coefficient $\lambda \in [0, 1]$ controls merge strength, where $\lambda = 0$ yields the Arabic update and $\lambda = 1$ yields the TIES-merged update. To match the original adapter budget, we refactorize $\Delta W_m^{(\text{merge})}$ back into rank- r LoRA form via truncated SVD. Non-selected projections/heads remain purely Arabic, which localizes cross-lingual injection to targeted attention components.

3.3 Reward Merging

We instantiate r_θ as a LoRA reward adapter on a frozen Aya Expanse 8B backbone, and extend cross-lingual merging to the reward-adapter space. We train Arabic and English reward adapters on Aya Expanse 8B and merge them using the same conflict-aware strategy. At inference time, we steer a frozen Aya Expanse 32B generator by fusing its next-token log-probabilities with the merged reward logits. Let $\log p_{\text{pt}}(y | x)$ denote the generator next-token

log-probabilities given prefix x , and let $r_\theta(x, y)$ denote the merged reward logit for token y . We form fused scores

$$v(x, y) = \log p_{\text{pt}}(y|x) + \alpha r_\theta(x, y),$$

and decode from $\text{softmax}(v)$ under shared generation constraints, and keep the generator frozen throughout.

4 Experimental Setup

4.1 Task and Data

We study abstractive Arabic news summarization on ARASUM [12]: given an Arabic article x , the model generates a short Modern Standard Arabic summary y (typically 1–2 sentences), using the editor-written lead as the training data. We create a fixed random 90/5/5 train/validation/test split from the $\sim 50\text{k}$ article–lead pairs; training uses the 90% split, model selection uses the 5% validation split, and all results are reported on the held-out 5% test split. To inject English supervision for cross-lingual transfer, we train English summarization and reward adapters on the non-anonymized CNN/DailyMail [13] training split ($\sim 287\text{k}$ examples), but evaluation remains strictly Arabic: inputs are Arabic articles and outputs are Arabic summaries, and English is used only through adapter/reward training and subsequent merging.

4.2 Models

Among open multilingual LLMs, the Aya Expanse family reports strong multilingual performance, including Arabic [14]. We use Aya Expanse 8B for all parameter-efficient training and merging, and use Aya Expanse 32B only as a frozen generator at inference time, steered by reward adapters trained on Aya Expanse 8B. We use the Aya Expanse tokenizer for both Arabic and English so all adapters remain in a shared parameter space. Inputs are formatted as instruction-style prompts followed by the article, i.e., “*Summarize the following Arabic text: {article} Summary:*”, with the lead as the target. We tokenize inputs to at most 1024 tokens and targets to 128 tokens.

4.3 Systems and Evaluation

We evaluate controlled baselines and ablations to isolate the effects of adapter merging and reward-guided decoding: (i) Aya-8B zero-shot; (ii) Aya-8B + Arabic LoRA; (iii) Aya-8B + merged LoRA (TIES/DARE-TIES); (iv) Aya-

32B zero-shot; (v) Aya-32B + PRT (Arabic reward); and (vi) Aya-32B + PRT (merged reward). All systems use shared generation settings.

For merged LoRA systems, we sweep (i) which attention projection(s) to merge (single $p \in \{q, k, v, o\}$ or selected pairs such as $q+v$, $k+v$, and $v+o$), (ii) head-wise localization (fraction of heads $\in \{1.0, 0.6, 0.4\}$), (iii) interpolation weight ($\lambda \in \{0.2, 0.3\}$), and (iv) trimming rate ($\rho \in \{0.20, 0.30\}$). We also evaluate DARE-TIES, which applies Drop-and-Rescale before TIES, and sweep the same settings. Full results are reported in the Appendix.

We apply no Arabic-specific normalization during training. For automatic evaluation, we apply a light, fixed normalization to both references and predictions before computing ROUGE and BLEU (e.g., removing tatweel/diacritics, normalizing common letter variants, and standardizing punctuation/whitespace). We compute BERTScore on the raw text. We report ROUGE-1/2/L, BLEU, and BERTScore [15], and also run pairwise LLM-as-a-judge A/B evaluation with GPT-5.1: given the article and two anonymized summaries in randomized order, the judge selects the better summary (or a tie) based on faithfulness, coverage, and Arabic fluency; we aggregate win/tie/loss counts. All results use deterministic decoding.

5 Results

We evaluate merging at two levels: (i) merging Arabic and English LoRA adapters on Aya Expanse 8B, and (ii) merging Arabic and English reward adapters to steer a frozen Aya Expanse 32B model under PRT. Full sweeps over projections, head subsets, and merge hyperparameters are reported in the Appendix. Although the AraSum test split contains roughly 2.5k article–lead pairs (5% of $\sim 50\text{k}$), we report results on a fixed random subset of 100 test articles for tractability.

5.1 LoRA Adapter Merging

Table 1 shows that in-language adaptation provides most of the gain: Aya-8B + Arabic LoRA improved over zero-shot on all metrics (+3.84 ROUGE-1, +1.33 ROUGE-2, +2.22 ROUGE-L, +1.59 BLEU, and +2.41 BERTScore). Cross-lingual merging yielded additional but smaller improvements over the Arabic LoRA baseline. The best single-projection TIES configuration achieved the highest BERTScore (73.74), while the best multi-projection

Table 1 Aya-8B baselines and best merged LoRA configurations on ARASUM (full sweeps in Appendix).

System	R-1	R-2	R-L	BLEU	BERT
Aya-8B zero-shot	21.06	5.91	16.26	2.99	70.65
Aya-8B + Arabic LoRA	24.90	7.24	18.48	4.58	73.06
Best TIES (Single)	25.34	7.55	18.65	4.66	73.74
Best TIES (Multiple)	25.14	7.34	18.44	5.05	73.50
Best DARE-TIES	25.89	7.71	19.41	4.75	73.67

Table 2 PRT-guided Aya-32B on ARASUM with Arabic-only vs. merged rewards (full sweeps in Appendix).

System	R-1	R-2	R-L	BLEU	BERT
Aya-32B zero-shot	23.54	7.26	17.43	4.74	72.15
Aya-32B + Arabic PRT	26.42	10.34	20.17	7.20	72.97
Best TIES (Single)	29.62	13.26	23.22	9.52	74.00
Best TIES (Multiple)	29.49	12.91	22.70	9.00	74.00
Best DARE-TIES	29.87	13.76	22.99	9.62	74.13

TIES configuration yielded the strongest BLEU (5.05). DARE-TIES merging attained the highest ROUGE-1/2 and ROUGE-L in Table 1, indicating that robustified merging can improve content selection while preserving Arabic quality.

Across the sweep, gains are most reliable when merging is localized (single projection and/or a restricted set of heads) and less reliable under broader multi-projection injection, consistent with our projection- and head-aware design.

5.2 Reward Adapter Merging

Table 2 shows that reward steering substantially improved a frozen Aya-32B generator. Relative to Aya-32B zero-shot, PRT with an Arabic-only reward increases performance by +2.88 ROUGE-1, +3.08 ROUGE-2, +2.74 ROUGE-L, and +2.46 BLEU, demonstrating that reward-only tuning can guide Aya-32B toward ARASUM-style summaries without updating generator weights. Merging reward adapters yields further gains beyond Arabic-only PRT: the best TIES merged reward reaches 29.62 ROUGE-1 and 74.00 BERTScore (and achieves the highest ROUGE-L), while DARE-TIES attains the best overall results in Table 2, with the highest ROUGE-1/2, BLEU, and BERTScore.

5.3 LLM-as-a-Judge

To complement automatic metrics, we run pairwise A/B evaluation using GPT-5.1 as the judge. For each article, the judge receives the source text and two anonymized sum-

maries in randomized order, and selects the better one (or a tie) based on faithfulness, coverage, and Arabic fluency.

Aya-8B LoRA merging. Cross-lingual merging is preferred to the Arabic-only LoRA baseline: the single-projection TIES merge is chosen more often than Arabic LoRA (44 vs. 34, with 22 ties). A value-projection DARE-TIES merge is also favoured over Arabic LoRA (50 vs. 44, with 6 ties). When compared directly, DARE-TIES is selected slightly more often than the single-projection TIES merge (44 vs. 40, with 16 ties).

Aya-32B PRT with merged rewards. Merging helps most reliably when the reward merge is focused: the single-projection TIES merged reward is preferred over Arabic-only PRT (46 vs. 22, with 32 ties). Broadening the merge to multiple projections yields a smaller advantage (46 vs. 38, with 16 ties), and the DARE-TIES merged reward is effectively tied with Arabic-only PRT (39 vs. 38, with 23 ties). Direct comparisons similarly favour the single-projection merged reward over both the multi-projection variant (46 vs. 35, with 19 ties) and the DARE-TIES merged reward (46 vs. 36, with 18 ties).

Cross-scale comparisons. Scale plus reward-guided steering dominates: the single-projection merged reward (PRT) beats Aya-8B + Arabic LoRA (58 vs. 39, with 3 ties) and also beats Aya-8B with DARE-TIES LoRA merging (66 vs. 31, with 3 ties). Aya-32B with Arabic-only PRT is likewise preferred over Aya-8B + Arabic LoRA (56 vs. 42, with 2 ties), suggesting a consistent cross-scale advantage.

6 Conclusion

We study cross-lingual model merging as a compute-efficient way to improve Arabic summarization without fully fine-tuning multilingual LLMs. On Aya Expanse 8B, head-wise, projection-aware TIES and DARE-TIES merges of Arabic and English LoRA adapters outperform an Arabic-only adapter on ARASUM across ROUGE, BLEU, BERTScore, and LLM-as-a-judge. At larger scale, we merge Arabic and English reward adapters and use the merged reward to steer a frozen Aya Expanse 32B model under PRT, beating zero-shot 32B and Arabic-only PRT without updating 32B weights. These findings indicate conflict-aware composition transfers English priors while preserving Arabic fluency and faithfulness. Future work will evaluate more datasets, merge multiple English sources and styles, and test other multilingual backbones.

Acknowledgement

This work was supported by JST CREST, Japan, Grant Number JPMJCR2565.

References

- [1] Mingxu Tao, Chen Zhang, Quzhe Huang, Tianyao Ma, Songfang Huang, Dongyan Zhao, and Yansong Feng. Unlocking the Potential of Model Merging for Low-Resource Languages. In **Findings of the Association for Computational Linguistics: EMNLP 2024**, p. 8705–8720. Association for Computational Linguistics, 2024.
- [2] Wissam Antoun, Fady Baly, and Hazem Hajj. AraBERT: Transformer-based Model for Arabic Language Understanding. In **Proceedings of the 4th Workshop on Open-Source Arabic Corpora and Processing Tools, with a Shared Task on Offensive Language Detection**. European Language Resources Association, 2020.
- [3] Moussa Kamal Eddine, Nadi Tomeh, Nizar Habash, Joseph Le Roux, and Michalis Vazirgiannis. AraBART: a Pre-trained Arabic Sequence-to-Sequence Model for Abstractive Summarization. In **Proceedings of the Seventh Arabic Natural Language Processing Workshop (WANLP)**, pp. 31–42. Association for Computational Linguistics, 2022.
- [4] Mo El-Haj and Paul Rayson. AraFinNews: Arabic Financial Summarisation with Domain-Adapted LLMs. **arXiv preprint arXiv:2511.01265**, 2025.
- [5] Lewis Tunstall, Leandro von Werra, and Thomas Wolf. **Natural Language Processing with Transformers**. O’Reilly Media, 2022.
- [6] Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-Rank Adaptation of Large Language Models. In **International Conference on Learning Representations (ICLR)**, 2022.
- [7] Shitao Xiao, Zheng Liu, Peitian Zhang, and Xingrun Xing. LM-Cocktail: Resilient Tuning of Language Models via Model Merging. In **Findings of the Association for Computational Linguistics: ACL 2024**, pp. 2474–2488. Association for Computational Linguistics, 2024.
- [8] Prateek Yadav, Derek Tam, Leshem Choshen, Colin Raffel, and Mohit Bansal. TIES-Merging: Resolving Interference When Merging Models. In **Thirty-seventh Conference on Neural Information Processing Systems**, 2023.
- [9] John Dang, et al. Aya Expand: Combining Research Breakthroughs for a New Multilingual Frontier. **arXiv preprint arXiv:2412.04261**, 2024.
- [10] Daiki Chijiwa, Taku Hasegawa, Kyosuke Nishida, Kuniko Saito, and Susumu Takeuchi. Portable Reward Tuning: Towards Reusable Fine-Tuning across Different Pretrained Models. In **Proceedings of the 42nd International Conference on Machine Learning**, pp. 10428–10448. PMLR, 2025.
- [11] 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 **Proceedings of the 41st International Conference on Machine Learning**, Vol. 235, pp. 57755–57775. PMLR, 2024.
- [12] Mram Kahla, Zijian Győző Yang, and Attila Novák. Cross-lingual Fine-tuning for Abstractive Arabic Text Summarization. In **Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2021)**, pp. 655–663, 2021.
- [13] Abigail See, Peter J. Liu, and Christopher D. Manning. Get To The Point: Summarization with Pointer-Generator Networks. In **Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)**, pp. 1073–1083. Association for Computational Linguistics, 2017.
- [14] Rawan Nasser Almatham, et al. BALSAM: A Platform for Benchmarking Arabic Large Language Models. In **Proceedings of The Third Arabic Natural Language Processing Conference**, pp. 258–277, Suzhou, China, 2025. Association for Computational Linguistics.
- [15] Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. BERTScore: Evaluating Text Generation with BERT. In **Proceedings of the 8th International Conference on Learning Representations (ICLR 2020)**, 2020.

A Appendix

This Appendix provides full sweeps for (i) adapter-level LoRA merging on Aya Expanse 8B and (ii) reward-level merging for steering a frozen Aya Expanse 32B generator under PRT, and reports the training hyperparameters used across all experiments.

A.1 Full Sweeps

Tables 3–6 report adapter-level merging on Aya-8B. **Proj** denotes the attention projection(s) merged (q, k, v, o), **Heads** is the fraction of heads merged per layer, λ controls merge strength, and ρ is the pruning rate.

Table 3 Baselines on ARASUM with Aya Expanse 8B.

System	R-1	R-2	R-L	BLEU	BERT
Aya-8B zero-shot	21.06	5.91	16.26	2.99	70.65
Aya-8B + Arabic LoRA	24.90	7.24	18.48	4.58	73.06

Table 4 Aya-8B: TIES single-projection LoRA merges.

Proj	Heads	λ	ρ	R-1	R-2	R-L	BLEU	BERT
v	1.0	1.0	0.20	24.77	7.31	18.43	4.28	73.58
o	1.0	1.0	0.20	24.83	7.12	18.56	4.91	73.31
q	1.0	1.0	0.20	24.21	6.70	18.07	4.28	73.42
k	1.0	1.0	0.20	24.79	7.23	18.48	4.97	73.43
v	0.5	0.5	0.20	24.57	6.90	18.80	3.97	73.52
o	0.5	0.5	0.20	25.03	7.26	18.60	4.99	73.36
q	0.5	0.5	0.20	24.69	7.31	18.16	4.70	73.17
k	0.5	0.5	0.20	24.73	7.26	18.48	4.94	73.37
v	0.6	0.6	0.20	25.22	7.08	18.46	4.97	73.40
o	0.6	0.6	0.20	24.83	7.18	18.41	4.94	73.37
q	0.6	0.6	0.20	25.03	7.38	18.45	4.98	73.42
k	0.6	0.6	0.20	24.98	7.27	18.55	4.94	73.36
v	0.8	1.0	0.30	25.14	7.18	18.72	4.47	73.50
o	0.8	1.0	0.30	24.72	7.06	18.41	4.78	73.29
q	0.8	1.0	0.30	24.72	7.63	18.33	5.06	73.73
k	0.8	1.0	0.30	24.95	7.22	18.45	5.00	73.42
v	0.4	0.4	0.20	25.34	7.55	18.65	4.66	73.74
o	0.4	0.4	0.20	24.78	7.16	18.47	4.85	73.31
q	0.4	0.4	0.20	25.15	7.29	18.41	4.93	73.30
k	0.4	0.4	0.20	24.94	7.25	18.48	4.96	73.36

Table 5 Aya-8B: TIES multi-projection LoRA merges.

Proj	Heads	λ	ρ	R-1	R-2	R-L	BLEU	BERT
q+v	1.0	1.0	0.20	24.93	7.36	18.69	4.86	73.25
q+k	1.0	1.0	0.20	24.42	7.31	18.33	4.72	73.10
k+v	1.0	1.0	0.20	24.87	7.31	18.73	4.92	73.32
v+o	1.0	1.0	0.20	24.67	7.24	18.46	4.46	73.51
q+v	0.6	0.6	0.20	24.95	7.39	18.36	5.02	73.17
q+k	0.6	0.6	0.20	24.88	7.23	17.98	4.90	73.28
k+v	0.6	0.6	0.20	25.14	7.34	18.44	5.05	73.50
v+o	0.6	0.6	0.20	25.09	7.03	18.23	4.88	73.44

Table 6 Aya-8B: DARE-TIES LoRA merges.

Proj	Heads	λ	ρ	R-1	R-2	R-L	BLEU	BERT
v	1.0	1.0	0.20	25.89	7.71	19.41	4.75	73.67
o	1.0	1.0	0.20	24.72	7.38	18.30	5.39	73.45
q	1.0	1.0	0.20	25.40	7.83	18.45	5.02	73.22
k	1.0	1.0	0.20	24.99	7.35	18.59	5.18	73.35
v	0.6	0.6	0.20	25.07	7.15	18.18	5.27	73.53
o	0.6	0.6	0.20	24.61	7.31	18.53	5.22	73.29
q	0.6	0.6	0.20	24.79	7.43	18.28	4.97	73.18
k	0.6	0.6	0.20	25.03	7.16	18.55	5.10	73.30
v	0.8	1.0	0.30	25.44	7.34	18.98	4.48	73.61
o	0.8	1.0	0.30	24.91	7.10	18.46	4.90	73.37
q	0.8	1.0	0.30	25.64	7.97	18.89	4.95	73.22
k	0.8	1.0	0.30	24.97	7.33	18.54	5.18	73.35

Tables 7–10 report reward-level merges used to steer a frozen Aya-32B generator via PRT, using the same notation as above.

Table 7 Baselines for Aya Expanse 32B with and without Arabic PRT reward.

System	R-1	R-2	R-L	BLEU	BERT
Aya-32B zero-shot	23.54	7.26	17.43	4.74	72.15
Aya-32B + Arabic-only reward	26.42	10.34	20.17	7.20	72.97

Table 8 Aya-32B: TIES single-projection PRT reward merges.

Proj	Heads	λ	ρ	R-1	R-2	R-L	BLEU	BERT
v	1.0	1.0	0.20	28.37	12.87	22.38	8.38	73.73
o	1.0	1.0	0.20	28.50	12.56	22.54	7.37	73.87
q	1.0	1.0	0.20	29.62	13.26	23.22	9.52	74.00
k	1.0	1.0	0.20	29.22	13.02	22.92	9.02	74.06
v	0.5	0.5	0.20	29.40	13.11	23.15	8.70	73.99
o	0.5	0.5	0.20	28.83	12.57	22.83	8.34	73.80
q	0.5	0.5	0.20	29.39	13.08	23.00	8.72	74.05
k	0.5	0.5	0.20	28.58	12.60	22.69	8.69	73.70
v	0.6	0.6	0.20	29.19	13.16	22.64	8.69	74.08
o	0.6	0.6	0.20	29.10	13.06	22.90	8.73	74.01
q	0.6	0.6	0.20	29.25	12.99	22.79	9.07	73.93
k	0.6	0.6	0.20	28.67	12.47	22.83	8.67	73.69
v	0.8	1.0	0.30	28.27	12.79	22.27	8.54	73.88
o	0.8	1.0	0.30	28.57	12.69	22.37	8.11	73.91
q	0.8	1.0	0.30	29.62	12.99	22.80	8.94	73.95
k	0.8	1.0	0.30	27.53	11.63	21.85	8.03	73.45
v	0.4	0.4	0.20	29.21	13.12	23.03	8.58	73.88
o	0.4	0.4	0.20	29.03	12.80	22.76	8.61	73.87
q	0.4	0.4	0.20	29.34	13.19	23.05	8.84	74.07
k	0.4	0.4	0.20	29.03	12.77	22.89	8.84	73.90

Table 9 Aya-32B: TIES multi-projection PRT reward merges.

Proj	Heads	λ	ρ	R-1	R-2	R-L	BLEU	BERT
q+v	1.0	1.0	0.20	27.82	11.82	21.72	8.09	73.43
q+k	1.0	1.0	0.20	28.94	12.87	23.05	9.06	73.91
k+v	1.0	1.0	0.20	28.87	13.22	23.10	9.01	73.79
v+o	1.0	1.0	0.20	28.93	12.29	22.44	8.64	73.92
q+v	0.6	0.6	0.20	29.49	12.91	22.70	9.00	74.00
q+k	0.6	0.6	0.20	28.82	12.33	22.48	8.38	73.77
k+v	0.6	0.6	0.20	27.75	12.01	21.56	8.27	73.62
v+o	0.6	0.6	0.20	27.49	12.11	21.81	8.47	73.52

Table 10 Aya-32B: DARE-TIES PRT reward merges.

Proj	Heads	λ	ρ	R-1	R-2	R-L	BLEU	BERT
v	1.0	1.0	0.20	29.28	13.11	22.58	8.86	73.95
o	1.0	1.0	0.20	28.90	12.57	22.64	9.05	74.10
q	1.0	1.0	0.20	28.94	12.39	22.44	8.70	73.70
k	1.0	1.0	0.20	29.23	12.86	23.04	8.84	73.85
v	0.6	0.6	0.20	29.77	13.82	23.08	9.62	73.92
o	0.6	0.6	0.20	29.68	13.83	23.17	9.62	73.93
q	0.6	0.6	0.20	28.96	12.53	22.31	9.41	73.87
k	0.6	0.6	0.20	29.71	13.02	22.96	8.87	73.70
v	0.8	1.0	0.30	29.87	13.76	22.99	9.62	74.13
o	0.8	1.0	0.30	28.87	12.67	22.22	9.41	74.05
q	0.8	1.0	0.30	29.68	12.96	23.00	8.87	74.04
k	0.8	1.0	0.30	28.51	12.44	22.68	8.41	73.64

A.2 Training Hyperparameters

We use LoRA rank $r=32$, scaling 64, and dropout 0.05, targeting attention projections, and train for three epochs with AdamW (learning rate 2×10^{-5} , weight decay 0.01), fp16, 500-step warmup, and linear decay, at effective batch size 16. We train LoRA reward adapters with comparable hyperparameters (learning rate 2×10^{-5} , three epochs, effective batch size 16), applying to attention projections.