

Lanigo at WMT26 Multilingual Instruction Shared Task (MIST): Full-Parameter Knowledge Distillation with Task-Conditional Pivot Decoding Under a 10B-Parameter Constraint

Adrian Charkiewicz, Artur Nowakowski

Lanigo, Poznań, Poland

{name}.{surname}@lanigo.com

Abstract

We describe our submission to the WMT26 Multilingual Instruction Shared Task (MIST). Our system is google/gemma-4-E4B-it (8B total / 4.5B effective parameters), fully fine-tuned via knowledge distillation from a larger teacher model (Gemma-4-31B-it) on teacher-generated responses spanning three sub-tasks and 24 languages. We find that (i) a two-judge quality-agreement filter on the distillation data provides no measurable benefit over using the teacher’s outputs unfiltered, at both LoRA and full-parameter training scale, and (ii) a task-conditional decoding strategy (using two-turn English drafting for open-ended generation and summarization, versus single-turn direct decoding for context-grounded question answering) significantly improves quality where applied and is harmless where it is not. Our final submission combines full-parameter distillation with this task-conditional decoding scheme.

1 Introduction

The WMT26 Multilingual Instruction Shared Task (MIST) evaluates a model’s ability to follow instructions across languages and task types under a constraint of at most 10B parameters, encouraging research on small, broadly-capable multilingual models (Kocmi et al., 2025).¹ The task covers three sub-tasks, specifically open-ended generation (OEG), context-grounded question answering (QA_CTX), and cross-lingual summarization (SUM), across 24 languages, including two low-resource languages (Yoruba, Central Kurdish) with limited native training data. Last year’s edition of this shared task received no valid participating systems (Kocmi et al., 2025), so there is little established submitted-system practice to compare against.

Our system fine-tunes google/gemma-4-E4B-it (Gemma Team, 2026) on teacher-distilled data

from a larger model in the same family, and decodes with a strategy chosen per sub-task (pivot vs. direct) rather than one mode applied uniformly. We report two findings we believe are useful beyond our own submission: a data-filtering ablation that is a clean negative result at two different training scales (§4.1), and a pivot-decoding result whose effect on quality is task-dependent and even sign-flips depending on whether the underlying model has been fine-tuned (§4.2). We also directly compare full fine-tuning against LoRA and SFT against CPO/KTO training objectives (§4.3, §4.4); a further ablation (per-task vs. combined-data models) is reported in Appendix B.

2 Related Work

Our study builds on four lines of work: knowledge distillation, preference-based fine-tuning objectives, task-conditional decoding, and LLM-as-judge evaluation.

Knowledge distillation from a larger teacher model into a smaller student (Hinton et al., 2015) lets us train a submission-eligible (≤ 10 B parameter) student on data quality closer to a much larger model’s capability. Unlike recent multi-teacher, cross-family distillation approaches for multilingual translation (Zheng et al., 2026), we use a single same-family teacher, matching the shared task’s single-model submission constraint.

Preference-based fine-tuning objectives, such as DPO (Rafailov et al., 2023), KTO (Ethayarajh et al., 2024), and CPO (Xu et al., 2024), offer natural alternatives to supervised fine-tuning (SFT) when contrastive (chosen/rejected) training signals are available. For our ablation experiments, preference pairs use the same AND-verified teacher candidate as SFT’s target for “chosen,” paired with the student’s own pre-fine-tuning response as “rejected” (restricted to prompts where that response was itself a judge-scored failure). We compare

¹WMT26 task page: <https://www2.statmt.org/wmt26/multilingual-instruction.html>.

these objectives directly (§4.4); while CPO showed a small, consistent edge over KTO specifically on QA_CTX, standard SFT matched or outperformed every preference-tuning variant overall. Consequently, contrastive objectives were omitted from our final submission, which relies exclusively on SFT.

Pivot / translate-then-generate prompting, where a model processes intermediate output in a high-resource language (typically English) before producing its final-language answer, has been explored as both a training-time technique (e.g., instruction tuning to internally pivot (Zhang et al., 2023)) and an inference-time strategy. Closer to our setting, we implement it purely at inference time as a two-turn chat strategy that reuses the first turn’s English draft as context, since model weights are fixed by the time decoding strategy is chosen. A recent study of pivot-augmented prompting for low-resource machine translation similarly finds modest, configuration-sensitive gains rather than uniform improvements (Ramasetu et al., 2026). This echoes our observation that pivot decoding’s effect is task-dependent and can even invert following fine-tuning, although their work focuses on translation quality via few-shot exemplar selection rather than our two-turn decoding framework for open-ended generation, QA, and summarization.

LLM-as-judge evaluation is necessary for scalable quality assessment across diverse multilingual tasks, where human evaluation is prohibitively expensive. Although automated judging carries known biases such as verbosity and self-preference (Gu et al., 2024), we mitigate same-family bias by employing both the teacher (Gemma-4-31B-it) and an independent cross-family evaluator (Qwen) as judges. Using both models allows us to cross-validate quality and test empirically for evaluator artifacts (for instance, Gemma wins base-model selection even under Qwen’s judge; §3.1). Finally, our two-judge agreement filter (§4.1) offers a lightweight alternative to iterative refine-then-judge data curation schemes (Cayir et al., 2025).

3 System Description

3.1 Base Model Selection

We evaluated two candidate base models, both satisfying the shared task’s ≤ 10 B-parameter constraint: google/gemma-4-E4B-it (8B total / 4.5B effective parameters) and Qwen3.5-9B (Qwen Team, 2026a) (a dense, hybrid-attention model,

~ 9.65 B parameters). On a stratified 1,200-row sample spanning all three sub-tasks (400 rows per task, which for open-ended generation covered virtually its entire dataset of just over 400 lines), judged by three independently-instantiated judge models under two decoding modes (extended-thinking and non-thinking), with a paired t -test pooling all judges,² we selected google/gemma-4-E4B-it over Qwen3.5-9B: it won in 5 of 6 judge $\times$ mode cells, including under Qwen’s own-family judge, with a pooled difference of +0.269 ($p = 1.1 \times 10^{-12}$) in non-thinking mode and +0.786 ($p = 1.7 \times 10^{-50}$) in thinking mode. Qwen3.5-9B’s thinking mode additionally showed a 23.7% empty-response/repetition-loop failure rate, posing a critical reliability concern independent of output quality. Consequently, we selected google/gemma-4-E4B-it as our foundational base model for all subsequent distillation and fine-tuning stages.

3.2 Training Data Construction

Training data is built by having a larger teacher model in the same family, Gemma-4-31B-it (Gemma Team, 2026), generate one response per prompt across all three sub-tasks and all task languages, for a pool of 20,625 (prompt, response) pairs. For Yoruba and Central Kurdish, which have no native task-formatted data, we backfill with general-purpose multilingual examples from the Aya Dataset (Singh et al., 2024); the volume and task-format match of this backfill varies by language.

We considered requiring two independently-instantiated judge models to agree that a candidate response is high quality before including it in training (an "AND-both" filter). §4.1 reports this ablation; we did not find measurable benefit and submit the unfiltered pool.

3.3 Training

The submitted model is fully fine-tuned on the full 20,625-pair unfiltered pool with the hyperparameters in Table 1. We compared this against per-task LoRA adapters and against CPO/KTO preference-optimization objectives trained on the same data (§4.3, §4.4); full-parameter SFT matched or out-

²"Qwen judge" throughout this paper refers to Qwen3.6-27B (Qwen Team, 2026b); "Gemma judge" refers to Gemma-4-31B-it (Gemma Team, 2026), the same checkpoint used as the distillation teacher; "Magistral judge" refers to Magistral-Small-24B (Mistral AI, 2025).

performed every alternative and was selected for submission.

Epochs	2
Learning rate	1×10^{-5}
LR schedule	cosine
Precision	bf16
Training pool	20,625 pairs

Table 1: Submitted model’s training hyperparameters.

3.4 Inference: Task-Conditional Decoding

At inference time we use task-conditional decoding: for OEG and SUM, the model first drafts a response in English before producing the final answer in the target language (pivot decoding); QA_CTX, the largest sub-task by volume (67.6% of the official test set), uses direct single-pass decoding. Appendix H details prompts; §4.2 reports ablations.

4 Experiments

4.1 Data-Filtering Ablation

Every candidate response is scored by each judge on a 1–7 Likert scale per rubric criterion (e.g. faithfulness, coherence; criteria vary by sub-task) and averaged; we call a response "high quality" if this average is $\geq 5/7$. "AND-both" keeps a training pair only if *both* Qwen and Gemma independently score it $\geq 5/7$; the alternative "unfiltered" condition keeps every teacher response regardless of score. We tested whether AND-both measurably improves training data quality over a single judge or no filter, on held-out per-task test data, Qwen-judge scored. We ran this comparison at *two* training scales. As Table 2 shows, the two-judge filter is not measurably load-bearing at either scale. We submit the unfiltered pool, which is simpler at equal measured quality.

4.2 Pivot vs. Direct Decoding

As shown in Table 3, the impact of pivot decoding depends heavily on both the target task and whether the model has been fine-tuned. While open-ended generation (OEG) and summarization (SUM) consistently benefit from pivot decoding across both base and fine-tuned checkpoints, context-grounded QA (QA_CTX) shows no performance gain after fine-tuning and degrades on the base model. Based on these empirical results, we adopt a task-conditional decoding strategy: applying pivot decoding to OEG and SUM while retaining direct decoding for QA_CTX.

Task (<i>n</i>)	Base	AND-both	Unfiltered
<i>LoRA Scale</i>			
OEG (42)	4.429	4.556	4.651
QA_CTX (1,484)	5.279	5.491	5.426
SUM (866)	5.991	6.161	6.152
<i>Full-FT Scale</i>			
OEG (42)	4.429	5.048	5.032
QA_CTX (1,484)	5.279	5.469	5.482
SUM (866)	5.991	6.237	6.209

Table 2: Data-filtering ablation at two training scales. Baseline scores correspond to the untrained base model.

Model	OEG	QA_CTX	SUM
Base	+0.857*	−0.408*	+0.132*
Fine-tuned	+1.238*	−0.053	+0.111*

Table 3: Pivot-decoding effect (pivot – direct), held-out data, Qwen-judge. * $p < 0.05$. On the untrained base model, pivot decoding is significantly *harmful* on QA_CTX ($p=1.3 \times 10^{-11}$) – terse English drafts get echoed back instead of translated, but this harm disappears entirely once the model is fine-tuned ($p=0.26$, n.s.), plausibly because SFT improves answer formatting enough to stop tripping the second-phase rewrite instruction. Pivot decoding helps OEG and SUM in both cases.

4.3 Full-Parameter Fine-Tuning (Full-FT) vs. LoRA

We compare full-parameter fine-tuning with LoRA to evaluate the practical performance gain relative to training cost for final system submission, rather than as a parameter-matched efficiency study. Both configurations are trained on the exact same dataset using the same SFT objective.

Task (<i>n</i>)	base	LoRA	full-FT
OEG (42)	4.429	4.556 (n.s.)	5.111*
QA_CTX (1,484)	5.279	5.491*	5.531*
SUM (866)	5.991	6.161*	6.274*

Table 4: Full fine-tuning vs. LoRA, both trained with the SFT objective. * $p < 0.05$ vs. base. Full-FT vs. LoRA directly: OEG +0.556 ($p = 0.062$, trend), QA_CTX +0.040 ($p = 0.22$, n.s.), SUM +0.114 ($p = 0.0028$, the only task where full-FT significantly beats LoRA). Full-FT is the only condition reaching significance over base on OEG, so we chose full-FT for submission.

4.4 Training Objective: SFT vs. CPO vs. KTO

We further evaluate full-parameter CPO and KTO preference-optimization objectives against SFT on the same per-task data split. To examine sensitivity to training stability, we compare two hyper-

Table 5: Hyperparameter setups for preference tuning.

Setup	Learning Rate	β (KL)	Init
SFT	2×10^{-5}	—	Base
Default	1×10^{-5}	0.1	Base / SFT
Tuned	2×10^{-6}	0.05	Base / SFT

parameter setups: a default baseline and a tuned regime with reduced learning rate and lower KL penalty (detailed in Table 5) across two initialization points: directly from the base model and from the fine-tuned SFT checkpoint.

Variant	OEG (42)	QA_CTX (1,484)	SUM (866)
base	4.429	5.279	5.991
CPO (default)	4.714	5.543*	6.278*
CPO (tuned, base)	4.333 [†]	5.495*	6.229*
CPO (tuned, SFT)	4.873	5.539*	6.303*
KTO (default)	4.476 [†]	5.465*	6.259*
KTO (tuned, base)	4.508	5.395*	6.303*
KTO (tuned, SFT)	5.111*	5.523*	6.261*
full-FT-SFT (chosen)	5.111*	5.531*	6.274*

Table 6: Comparison of training objectives (all full-parameter). “default/tuned” = default vs. tuned hyperparameters; “base/SFT” = initialization checkpoint. * $p < 0.05$ vs. base; [†] $p < 0.05$ vs. SFT (worse). While CPO and KTO improve over base on most tasks, no variant significantly outperforms full-FT-SFT.

5 Results

5.1 Submitted Model vs. Untrained Base

On the official 12,775-row test set, pairwise comparison (submitted model vs. the untrained base model, task-conditional decoding on both sides) under two independent judges shows the submitted model winning decisively (Table 7).

Judge	Submitted wins	Base wins	Ties
Qwen	3,940	2,408	6,427
Gemma	3,038	2,296	7,441

Table 7: Official test-set pairwise comparison, submitted model vs. untrained base, task-conditional decoding on both sides.

Because Gemma-4-31B-it also served as the distillation teacher, performance gains observed exclusively under the Gemma judge could partly reflect self-preference bias—favoring outputs derived from its own generations—rather than genuine quality improvements. Since Qwen generated no training data, we treat it as an independent evaluator. To ensure our results hold regardless of judge selection, we evaluate the submitted and base

Task (n)	Qwen Judge		Gemma Judge	
	Δ	d	Δ	d
OEG (42)	+0.603*	0.33	+0.270	n.s.
QA_CTX (1,484)	+0.204*	0.12	+0.181*	0.12
SUM (866)	+0.218*	0.16	+0.086*	0.10

Table 8: Per-task score difference (Δ = submitted – base) and effect size (d) across Qwen and Gemma judges using paired t -tests (* $p < 0.05$). Fine-tuning gains remain statistically significant across both judges on QA_CTX and SUM.

model outputs on the held-out test set under both judges using a paired t -test.

The effect holds under both judges on QA_CTX and SUM, so it is not an artifact of Gemma’s self-preference risk; OEG’s smaller, non-significant effect is consistent in direction and attributable to its small held-out size ($n = 42$), matching the pattern already seen for OEG elsewhere in this paper.

5.2 Distillation vs. Unadapted Gold Data

To evaluate whether teacher distillation is necessary, we compare our submitted model against an otherwise identical full-parameter checkpoint trained on the shared task’s reference (gold) text. Because raw reference targets often differ in formatting style (e.g., terse spans versus verbose task-formatted responses), we evaluate both verbatim reference text and a reformatted variant where the teacher rewrites each reference into the target format while preserving underlying content.

On the official test set, the distilled submitted model wins decisively over the verbatim reference baseline across both judges (Table 9). When evaluating the reformatted reference model on held-out data under the Qwen judge, open-ended generation (OEG) shows a significant advantage for reference training over the distilled model ($+1.333$, $d = 0.69$, $p = 6.7 \times 10^{-5}$), a moderate-to-large effect comparable in size (though opposite in direction) to the SUM finding. Context-grounded QA (QA_CTX) yields comparable performance. However, summarization (SUM) consistently favors the distilled model ($d = -0.65$).

We attribute this residual gap on SUM to the inherent noise in retrieved cross-lingual reference corpora such as CrossSum (Hasan et al., 2021), where targets are aligned via semantic similarity rather than direct translation. In contrast, teacher distillation ensures that target summaries are generated directly from the input context, resulting in tighter

source grounding and fewer factual hallucinations. Complete comparisons, including reformatted reference results and per-source breakdowns, are detailed in Appendix A.

Judge	Reference wins	Submitted wins	Ties
Qwen	1,488	5,980	5,307
Gemma	1,418	5,489	5,868

Table 9: Pairwise comparison on the official test set between the verbatim reference-trained model and the submitted distilled model (task-conditional decoding applied to both). The submitted model outperforms the reference baseline under both judges.

5.3 Low-Resource Languages: Yoruba and Central Kurdish

Yoruba and Central Kurdish lack native task-formatted training data (§3.2). Fine-tuning yields minimal gains for both on QA_CTX, with Yoruba remaining at a floor score of 1.0/7. However, pivot decoding reveals stark differences between them: while direct decoding was chosen globally for QA_CTX due to a null aggregate effect (-0.053), applying pivot decoding to Yoruba recovers $+1.500$ points ($p < 0.05$, Table 10). Central Kurdish shows no such gain (-0.222 , n.s.). This divergence demonstrates that low-resource languages do not behave uniformly, indicating that per-language decoding strategies may be beneficial.

	n	FT-base	Pivot-direct
Yoruba (QA_CTX)	32	0.000	$+1.500^*$
Yoruba (SUM)	18	$+0.222$	$+1.250^*$
Central Kurdish (QA_CTX)	27	$+0.222$	-0.222
All languages (QA_CTX)	1,484	$+0.204^*$	-0.053
All languages (SUM)	866	$+0.218^*$	$+0.111^*$

Table 10: Held-out low-resource performance (Qwen judge). “FT-base” is fine-tuning gain; “Pivot-direct” is pivot-decoding effect. $^*p < 0.05$. Central Kurdish lacks held-out SUM data.

6 Limitations

Our submission operates under the WMT26 MIST $\leq 10\text{B}$ -parameter constraint, which rules out larger architectures. Additionally, Yoruba and Central Kurdish lack native task-formatted data and rely on general-purpose Aya Dataset examples (Singh et al., 2024), whose format does not fully match the target distribution (§5.3). While our reference-data diagnosis isolated a major defect in summariza-

tion (§5.2), fine-grained error analyses were not extended to QA_CTX.

Furthermore, using Gemma-4-31B-it as both the distillation teacher and an evaluation judge introduces potential self-preference bias; future work should validate these findings with human evaluation, particularly as the reliability of LLM judges on low-resource languages remains an open question. Methodologically, our 5/7 filtering threshold was a heuristic driven by compute constraints, and filtering was not compared against a size-matched random baseline. Finally, while pivot decoding improves aggregate quality, generating an English draft adds an inference step that increases latency, and risks stripping away localized idioms and cultural nuances in the target languages, particularly in open-ended generation (OEG).

7 Conclusion

Our WMT26 MIST submission fine-tunes google/gemma-4-E4B-it on unfiltered teacher-distilled targets from Gemma-4-31B-it, paired with a task-conditional decoding strategy (pivot decoding for OEG and SUM; direct decoding for QA_CTX). Beyond overall system performance, our analysis highlights two key empirical insights: (1) quality-based data filtering yields a consistent negative result across training scales; and (2) the effectiveness of pivot decoding depends heavily on task type and fine-tuning setup.

References

- Derin Cayir, Renjie Tao, Rashi Rungta, Kai Sun, Sean Chen, Haidar Khan, Minseok Kim, Julia Reinspach, and Yue Liu. 2025. [Refine-n-judge: Curating high-quality preference chains for LLM-fine-tuning](#). *arXiv preprint arXiv:2508.01543*.
- Kawin Ethayarajh, Winnie Xu, Niklas Muennighoff, Dan Jurafsky, and Douwe Kiela. 2024. [KTO: Model alignment as prospect theoretic optimization](#). *arXiv preprint arXiv:2402.01306*.
- Gemma Team. 2026. [Gemma 4 technical report](#). *Preprint*, arXiv:2607.02770.
- Jiawei Gu, Xuhui Jiang, Zhichao Shi, Hexiang Tan, Xuehao Zhai, Chengjin Xu, Wei Li, Yinghan Shen, Shengjie Ma, Honghao Liu, Saizhuo Wang, Kun Zhang, Yuanzhuo Wang, Wen Gao, Lionel Ni, and Jian Guo. 2024. [A survey on LLM-as-a-judge](#). *arXiv preprint arXiv:2411.15594*.
- Tahmid Hasan, Abhik Bhattacharjee, Wasi Uddin Ahmad, Yuan-Fang Li, Yong-bin Kang, and Rifat Shahri-

- yar. 2021. [Crosssum: Beyond English-centric cross-lingual abstractive text summarization for 1500+ language pairs](#). *arXiv preprint arXiv:2112.08804*.
- Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. 2015. [Distilling the knowledge in a neural network](#). *arXiv preprint arXiv:1503.02531*.
- Tom Kocmi, Ekaterina Artemova, Eleftherios Avramidis, Eleftheria Briakou, Pinzhen Chen, Marzieh Fadaee, Markus Freitag, Roman Grundkiewicz, Yupeng Hou, Philipp Koehn, Julia Kreutzer, Saab Mansour, Stefano Perrella, Lorenzo Proietti, Parker Riley, Eduardo Sánchez, Patricia Schmidova, Mariya Shmatova, and Vilém Zouhar. 2025. [Findings of the WMT25 multilingual instruction shared task: Persistent hurdles in reasoning, generation, and evaluation](#). In *Proceedings of the Tenth Conference on Machine Translation*, pages 414–435, Suzhou, China. Association for Computational Linguistics.
- Mistral AI. 2025. Magistral-small-24b. <https://huggingface.co/mistralai>. Open-weights LLM.
- Qwen Team. 2026a. [Qwen3.5: Towards native multi-modal agents](#).
- Qwen Team. 2026b. [Qwen3.6-27B: Flagship-level coding in a 27B dense model](#).
- Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D. Manning, Stefano Ermon, and Chelsea Finn. 2023. [Direct preference optimization: Your language model is secretly a reward model](#). In *Advances in Neural Information Processing Systems 36 (NeurIPS 2023)*.
- Aishwarya Ramasethu, Rohin Garg, Niyathi Allu, Harshwardhan Fartale, and Dun Li Chan. 2026. [Can linguistically related languages guide LLM translation in low-resource settings?](#) In *Proceedings for the Ninth Workshop on Technologies for Machine Translation of Low Resource Languages (LoResMT 2026)*, pages 168–185, Rabat, Morocco. Association for Computational Linguistics.
- Shivalika Singh, Freddie Vargus, Daniel D’souza, Börje F. Karlsson, Abinaya Mahendiran, Wei-Yin Ko, Herumb Shandilya, Jay Patel, Deividas Mataciunas, Laura O’Mahony, Mike Zhang, Ramith Het-tiarachchi, Joseph Wilson, Marina Machado, Luisa Moura, Dominik Krzemiński, Hakimeh Fadaei, Irem Ergun, Ifeoma Okoh, and 14 others. 2024. [Aya dataset: An open-access collection for multilingual instruction tuning](#). In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11521–11567, Bangkok, Thailand. Association for Computational Linguistics.
- Haoran Xu, Amr Sharaf, Yunmo Chen, Weiting Tan, Lingfeng Shen, Benjamin Van Durme, Kenton Murray, and Young Jin Kim. 2024. [Contrastive preference optimization: Pushing the boundaries of LLM performance in machine translation](#). In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 55204–55224. PMLR.
- Zhihan Zhang, Dong-Ho Lee, Yuwei Fang, Wenhao Yu, Mengzhao Jia, Meng Jiang, and Francesco Barbieri. 2023. [PLUG: Leveraging pivot language in cross-lingual instruction tuning](#). *arXiv preprint arXiv:2311.08711*.
- Mao Zheng, Zheng Li, Tao Chen, Bo Lv, Mingrui Sun, Mingyang Song, Jinlong Song, Hong Huang, Decheng Wu, Hai Wang, Yifan Song, Yanfeng Chen, and Guanwei Zhang. 2026. [Hy-MT2: A family of fast, efficient and powerful multilingual translation models in the wild](#). *arXiv preprint arXiv:2605.22064*.

A Distillation vs. Reference Training: Full Comparison

Section 5.2 provides a high-level summary of the comparison between teacher distillation and direct reference training. Here we detail the complete methodology, experimental setup, and diagnostic evaluations.

To evaluate whether teacher distillation is strictly necessary, we trained an otherwise identical full-parameter checkpoint on the dataset’s gold reference outputs using the same training recipe and prompt pool (20,625 examples). Only 0.2% of the teacher-distilled training target strings are identical to their corresponding reference text, ensuring this comparison measures the effect of the target distribution rather than near-duplicate memorization.

In our initial baseline experiment, reference targets were passed verbatim without reformatting to match our per-language instruction prompt. Open-ended generation (OEG) reference targets were generated by GPT-4.1 for the WMT25 benchmark (Kocmi et al., 2025), representing a high-quality model output. Conversely, summarization (SUM) and context-grounded QA (QA_CTX) targets were drawn from pre-existing datasets (e.g., CrossSum, WikiLingua, Belebele, TyDiQA) that were never formatted for instruction-following prompts, often consisting of terse text spans (e.g., “3: London”). To address this format mismatch, we later introduced a reformatted reference setup.

To test whether combining reference and teacher targets yields further improvements, we constructed a “best-of” composite dataset. For OEG and SUM, reference outputs were scored as model responses using the evaluator judge, and the higher-scoring target (teacher vs. reference) was selected for each prompt. Because the QA_CTX evaluation

rubric checks whether an answer matches the reference, comparing the reference to itself produces a circular signal; thus, for QA_CTX, we retained the teacher response if its evaluation score was $\geq 5/7$, falling back to the reference target otherwise. Table 11 reports the selection rates across tasks.

Task	Reference Chosen	Selection Rule
OEG	69.4% (263/379)	Head-to-head evaluation
SUM	4.7% (356/7,526)	Head-to-head evaluation
QA_CTX	20.7% (2,583/12,457)	Floor score rule ($< 5/7$)
Overall	16.8% (3,461/20,625)	—

Table 11: Proportion of reference targets selected over teacher responses in the composite “best-of” dataset. SUM’s low reference selection rate stems from reference targets scoring below 5/7 on 44% of head-to-head comparisons due to formatting discrepancies.

As shown in Table 12, the composite best-of model performs on par with the submitted teacher-distilled model rather than exceeding it. Because the teacher response was retained whenever it matched or exceeded the reference score, 83.2% of the best-of training pool remains identical to the teacher-only dataset. For the remaining 16.8%, SUM swapped only 4.7% of its rows, while OEG underwent a major shift toward reference targets (69.4%) but showed no statistically significant difference on held-out evaluation (Table 13). Consequently, instance-level target selection largely recovers the pure teacher-distilled policy.

Fixing the Format Confound. To isolate dataset quality from presentation artifacts, we created a reformatted reference dataset. For SUM and QA_CTX, the teacher model rewrote each reference into a full, instruction-compliant response while preserving all factual content and introducing no new information (e.g., converting a terse span like “3: London” into a complete descriptive sentence). Manual inspection confirmed that rewrites maintained strict factual fidelity across languages. We then fine-tuned a model on this reformatted reference set and re-evaluated performance on the held-out set (Table 14).

Reformatting reference targets significantly improves performance over raw reference training across OEG and SUM. On QA_CTX, performance reaches a tie with the submitted model. However, on SUM, a substantial gap remains ($d = -0.65$), indicating that format mismatch alone does not account for the entirety of the performance difference.

Task (n)	vs. Submitted	vs. Reference-only
OEG (42)	+0.429	+0.016
QA_CTX (1,484)	+0.027	+0.004
SUM (866)	−0.014	+1.750*

Table 12: Held-out per-task effect of the best-of-{teacher, reference} model (mean difference under the primary Qwen judge; $*p < 0.05$). The best-of model does not differ significantly from the teacher-only submitted model on any task, with its primary gain appearing over raw reference training on SUM ($d = 0.87$).

Task (n)	Score Difference (Submitted − Reference)
OEG (42)	+0.413 (n.s.)
QA_CTX (1,484)	+0.023 (n.s.)
SUM (866)	+1.764* ($d=0.88$)

Table 13: Held-out per-task performance difference between teacher-distilled submitted and raw reference-trained models under the Qwen judge ($*p < 0.05$). SUM demonstrates a large advantage for the teacher-distilled model ($p = 2.65 \times 10^{-110}$).

Residual Length Gap and CrossSum Misalignment. Even after reformatting, reference targets for SUM remain substantially shorter than teacher-distilled outputs (66.9 vs. 165.3 average words; QA_CTX shows 27.2 vs. 74.8 words), as the reformatting prompt instructed the teacher to add no extraneous information beyond the gold reference. Examining per-criterion scores for SUM reveals that the gap is concentrated primarily in coverage (6.41 vs. 3.46), whereas faithfulness (6.22 vs. 5.38) and naturalness (6.06 vs. 5.84) exhibit smaller differences.

Further source breakdown (Table 15) reveals that the performance degradation in SUM is concentrated within CrossSum (Hasan et al., 2021), which comprises 77.9% of the reference pool. CrossSum aligns articles across languages using embedding similarity rather than verified translations, causing 23.9% of its rows to contain reference summaries with factual assertions unsupported by the source text (e.g., a source text detailing a 2–1 football match paired with a reference summary claiming a 3–0 outcome). On clean reference subsets (WikiLingua and MCIF, $n = 177$), the overall performance gap shrinks to non-significance (+0.167, $p = 0.13$). Because our submitted model is fine-tuned on synthetic teacher targets generated directly from each input source, it avoids these alignment defects.

Task (n)	vs. Submitted	vs. Raw Reference
OEG (42)	+1.333* ($d=0.69$)	+0.921* ($d=0.44$)
QA_CTX (1,484)	+0.069 (n.s.)	+0.046 (n.s.)
SUM (866)	-1.293* ($d=-0.65$)	+0.472* ($d=0.28$)

Table 14: Held-out per-task effect of reformatted reference training under the Qwen judge ($*p < 0.05$). Reformatting significantly improves over raw reference training on OEG and SUM, narrowing the gap with the submitted model on SUM from $d = -0.88$ to $d = -0.65$.

Source Dataset (n)	Ref. Floor Rate	Δ Faithfulness	Δ Coverage
CrossSum (689)	23.9%	+0.95*	+3.57*
WikiLingua (148)	4.7%	+0.43*	+0.53*
MCIF (29)	0.0%	+0.34 (n.s.)	+0.14 (n.s.)

Table 15: Detailed breakdown of the SUM reference-training gap by source dataset under the Qwen judge. “Ref. Floor Rate” denotes the proportion of outputs scoring 1/7 on both faithfulness and coverage. Δ columns represent submitted minus reference scores ($*p < 0.05$).

B Per-Task Adapters vs. Combined-Data Model

To evaluate task-specific capacity allocation, we compared training separate LoRA adapters for each sub-task against training a single full-parameter model on all combined tasks. Evaluation on held-out data using full-parameter fine-tuning reveals no statistically significant differences between single-task and multi-task setups (Table 16).

Task (n)	Per-Task Model	Combined Model	p -value
OEG (42)	5.111	5.000	0.68 (n.s.)
QA_CTX (1,484)	5.531	5.495	0.22 (n.s.)
SUM (866)	6.274	6.235	0.28 (n.s.)

Table 16: Performance comparison between per-task adapters and a single combined-data model under full-parameter fine-tuning. All differences are non-significant.

Given the equivalent empirical performance, we deploy a single combined-data model to streamline model serving and inference deployment.

C Base Model Selection Breakdown

Table 17 and Table 18 present the complete breakdown of the base model comparison across evaluation judges and decoding modes summarized in Section 3.1.

Enabling extended thinking improves Gemma across all tasks (+0.251 to +1.307). In contrast, Qwen exhibits mixed behavior: extended thinking improves QA_CTX (+0.390, $p=7.1 \times 10^{-8}$) but severely degrades SUM (-0.612 , $p=1.4 \times 10^{-10}$).

Diagnostic inspection revealed that Qwen’s thinking mode encountered runaway enumeration failures on 23.67% (284/1,200) of rows, producing empty final responses. Increasing the generation token budget from 3,072 to 6,144 tokens resolved only a minor fraction of these errors. Gemma produced zero empty responses across all 1,200 evaluation runs.

D Full Data-Filtering Ablation Across Judge Conditions

Table 19 details the full data-filtering ablation comparing single-judge filtering setups (gemma-only, qwen-only), dual-judge intersection filtering (AND-both), and unfiltered training across LoRA and full-parameter scales.

At full-parameter scale, filtering targets via judge scores yields no statistically significant performance gains over unfiltered teacher distillation, establishing a clear negative result for quality-based data filtering when using strong teacher models.

E Pivot-Decoding Robustness Across Checkpoints

To verify whether the interaction between fine-tuning and pivot decoding is robust across training stages, we evaluated an intermediate full-parameter checkpoint trained prior to final dataset refinement (§3.2).

As shown in Table 20, the sign-flip/attenuation behavior on QA_CTX appears in both fine-tuned checkpoints, confirming that the change in pivot decoding efficacy is a systematic property of task-specific fine-tuning.

F Engineering and Infrastructure Diagnostics

During system development, two major engineering issues were identified and resolved:

Cross-Layer KV Sharing Checkpoint Loading in vLLM. The final 18 layers of gemma-4-E4B-it omit dedicated key/value projections to utilize cross-layer KV sharing. While Hugging Face’s `Trainer.save_model()` exports these parameter structures natively, vLLM’s checkpoint loader expects explicit key/value weight matrices for every layer, leading to loading failures. We resolved this by applying a post-processing script that constructs duplicate weight references

Variant	Qwen Judge	Gemma Judge	Magistral Judge	Cross-Judge Median
gemma_thinking	6.537	6.613	6.727	6.613
gemma_no_thinking	5.521	6.202	6.619	6.202
qwen_no_thinking	5.184	5.764	6.586	5.764
qwen_thinking	5.610	5.416	6.492	5.610

Table 17: Overall average score per evaluator judge across candidate base models. Gemma-4-E4B-it achieves superior performance across all judge families.

Mode	Task	Gemma – Qwen	Cohen’s d	95% CI	p -value
no_thinking	OEG	+0.251	0.22	[0.14, 0.36]	1.8×10^{-5}
no_thinking	QA_CTX	+0.245	0.15	[0.08, 0.41]	3.2×10^{-3}
no_thinking	SUM	+0.312	0.31	[0.21, 0.41]	1.0×10^{-9}
no_thinking	Pooled	+0.269	0.21	[0.20, 0.34]	1.1×10^{-12}
thinking	OEG	+1.005	0.57	[0.83, 1.18]	2.6×10^{-26}
thinking	QA_CTX	+0.047	0.04	[−0.08, 0.17]	0.46 (n.s.)
thinking	SUM	+1.307	0.70	[1.12, 1.49]	3.7×10^{-36}
thinking	Pooled	+0.786	0.45	[0.69, 0.88]	1.7×10^{-50}

Table 18: Paired t -test comparison between Gemma-4-E4B-it and Qwen3.5-9B across identical prompts ($n = 400$ per task; $n = 1,200$ pooled).

Task (n)	AND [†]	Unfilt.	Gemma [‡]	Qwen [§]
<i>LoRA Scale</i>				
OEG (42)	4.556	4.651	4.778	4.667
QA_CTX (1,484)	5.491	5.426	5.469	5.462
SUM (866)	6.161	6.152	6.174	6.208
<i>Full-FT Scale</i>				
OEG (42)	5.048	5.032	4.873	4.794
QA_CTX (1,484)	5.469	5.482	5.511	5.478
SUM (866)	6.237	6.209	6.239	6.241

Table 19: Held-out performance under four data-filtering regimes across LoRA and full-parameter scales (Qwen judge). Regimes: [†]AND-both, [‡]gemma-only, [§]qwen-only. Baseline base scores: OEG 4.429, QA_CTX 5.279, SUM 5.991. Pairwise differences at full-FT scale are non-significant ($p \in [0.28, 0.94]$).

in an auxiliary safetensors shard, enabling standard deployment within vLLM.

Low-Resource Abstention Behavior on Misformatted Data. On an initial dataset iteration using raw Aya backfill targets, the QA_CTX abstention rate for Yoruba reached 73.0%, significantly exceeding the 20.0–26.0% rate observed across other languages. Re-tuning after standardizing system prompts and target delimiters across the Yoruba backfill reduced its abstention rate to 18.6%, aligning it with the overall cross-lingual average (9.7–27.2%).

Model Checkpoint	OEG	QA_CTX	SUM
Base Model	+0.857*	−0.408*	+0.132*
Fine-tuned (Intermediate Checkpoint)	+0.778*	+0.020 (n.s.)	+0.070 (n.s.)
Fine-tuned (Submitted Checkpoint)	+1.238*	−0.053 (n.s.)	+0.111*

Table 20: Pivot-decoding effect (Pivot – Direct) across three checkpoints ($*p < 0.05$). The shift on QA_CTX from negative on the base model to neutral on fine-tuned checkpoints is consistent across both fine-tuned models.

G Optimization Dynamics and Shortcut Learning in Preference Tuning

In early development, we experimented with training DPO and KTO on a failure-biased subset ($\sim 1.5k$ prompts where the base model produced incorrect answers). Candidates generated by teacher models or reference outputs were designated as “chosen,” while the base model’s failed outputs served as “rejected.”

Models trained under this setup consistently degraded relative to the untrained base model (dev-set win rate 16.9–18.5%; Wilcoxon $p = 4.3 \times 10^{-5}$). A control experiment with randomly shuffled chosen/rejected pairs resulted in severe policy collapse ($p = 5.5 \times 10^{-11}$). Evaluating an on-policy setup using self-sampled outputs (best-of-8 under the judge) similarly yielded negative results on OEG (-0.183 , $p = 0.010$) and neutral performance on QA_CTX and SUM.

Training logs (Table 21) revealed that chosen responses were consistently concise, whereas rejected responses were verbose and hedged (Ta-

Training Epoch	Train Reward Accuracy	Reward Margin
0.08	39.0% (Chance)	−0.005
0.32	99.0%	+0.370
0.73	100.0%	+4.530
2.00 (Final)	100.0%	+6.830

Table 21: DPO reward metrics over training on failure-biased data (989 pairs). The reward margin saturates by 0.73 epochs without further task-learning gains.

ble 22).

Recipe	Chosen Median Words	Rejected Median Words	Length Ratio
Teacher-Verified	63	106	1.55–4.0×
Best-of-Both (QA_CTX)	—	—	Up to 33×

Table 22: Median word count asymmetry between chosen and rejected sets. Base model failure outputs exhibit consistent verbosity relative to external targets.

Rather than learning semantic correctness, the preference objectives fit a simple length shortcut (“favor shorter, assertive responses”), saturating reward metrics rapidly while degrading model capability on broader distributions.

Checkpoint	OEG	QA_CTX	SUM
DPO (Epoch 0.48)	+0.022 (n.s.)	+0.017 (n.s.)	−0.060*
DPO (Epoch 0.85)	+0.067*	+0.033 (n.s.)	−0.056 (n.s.)
Plain SFT	+0.011 (n.s.)	+0.067 (n.s.)	−0.125*

Table 23: Evaluation of early stopping and non-contrastive SFT on a 300-row held-out sample (* $p < 0.05$).

As shown in Table 23, fine-tuning on the narrow failure slice using SFT instead of DPO still failed to recover performance on SUM. To address this, we transitioned to SimPO/CPO and expanded the training set to a 90/10 task-wide data blend. This combination resolved the performance loss on SUM, yielding the setup detailed in Section 4.4.

H Prompts and Generation Settings

For OEG and SUM, we apply a two-turn pivot decoding strategy. In the first turn, the model receives the row’s original task prompt with the following instruction override appended, which forces an English draft regardless of the language requested: *“No matter what language is requested above, write your ENTIRE answer in English for this step. Everything else about the task (content, format, completeness) stays exactly the same.”*

The resulting English draft is appended to the conversation as an assistant turn, and the model is given a second, standalone user turn: *“Now give*

your final answer. Rewrite the English draft above in {language}, preserving its content, meaning, and format exactly. Respond only in {language} – do not include any English.” where {language} is substituted with the full English name of the target language. No system prompt is used.

For QA_CTX, we use a single-turn direct prompt: the model receives only the original task prompt, with neither the override nor a second turn. To isolate the learned policy, all generations across both training and inference stages use greedy decoding (temperature=0.0) with a maximum generation limit of max_new_tokens=512 per turn (capping pivot rows at 1,024 tokens total).

I Inference Cost of Pivot Decoding

Pivot decoding requires two sequential generation calls per row instead of one. We measured the realized token overhead on held-out data from the submitted model (Table 24). We did not measure wall-clock latency directly, but the token-volume and call-count overhead make the practical cost of pivot decoding explicit.

Task	Direct	Pivot (Draft + Final)	Overhead
OEG	205.8	285.0 + 195.6 = 480.6	2.3×
SUM	160.9	173.7 + 147.7 = 321.4	2.0×

Table 24: Token-volume overhead of pivot decoding vs. direct decoding (average words per response).

J Extended LLM-as-Judge Diagnostics

Judge Disagreement on OEG. Gemma and Qwen judges disagree substantially more often on OEG pass/fail rates than on QA_CTX or SUM (86–99% agreement on the latter). This is consistent with self-preference risk being sharpest on the least-constrained task.

Low-Resource Judge Reliability. On the base-model evaluation sample ($n = 16$ rows per language), Gemma and Qwen judges show strong rank agreement on both Yoruba and Central Kurdish (Pearson $r = 0.95$ – 0.99 in non-thinking mode), at least as strong as the cross-language average ($r = 0.73$ – 0.84). Absolute score gaps are occasionally larger for Yoruba specifically (mean absolute difference = 1.34 in thinking mode, vs. 0.21 overall), though both judges agree the outputs are low-quality (mean scores 2.3–3.8/7) rather than disagreeing on direction. Because this sample is small

and drawn from an earlier untrained stage, we treat this as suggestive rather than conclusive.