

CUNI-UFAL at WMT26 Multilingual Instruction Shared Task

Aleš Manuel Papáček and Patrícia Schmidtová*

Institute of Formal and Applied Linguistics
Faculty of Mathematics and Physics, Charles University
Prague, Czech Republic
{papacek, schmidtova}@ufal.mff.cuni.cz

Abstract

We describe our submission to the WMT26 Multilingual Instruction Shared Task, which evaluates sub-10B-parameter models on multilingual question answering, open-ended generation, and cross-lingual summarization. We compare four open-weight models and vary the system prompt, reasoning mode, and decoding parameters. On our development set, the tested prompt and reasoning changes have a larger effect on LLM-judge scores than the difference between the two finalist models in their best settings. Gemma 4 E4B obtains the highest overall judge score, whereas Qwen 3.5 9B follows hard constraints more reliably on our internal benchmark. We therefore submit Qwen as our primary system and Gemma as a secondary system. We also introduce a 1,283-example diagnostic benchmark for grounded QA, constrained generation, and abstract writing from ACL papers in 24 languages. Using it for QLoRA fine-tuning does not work in our setup: it lowers validation loss but does not improve quality under our LLM judge. In the official automatic evaluation, our Gemma system ranks first overall and our Qwen system second; Qwen wins QA-context, while Gemma wins QA-OEG and places second in summarization.

1 Introduction

The WMT26 Multilingual Instruction Shared Task (MIST) (Schmidtová et al., 2026) evaluates language models with fewer than 10B parameters on question answering, open-ended generation, and cross-lingual summarization. Systems must answer in 24 languages while following content and formatting constraints. A previous edition was held at WMT25 with a slightly different set of tasks (Kocmi et al., 2025).

We test four open-weight models (Qwen 3.5 9B (Qwen Team, 2026), Gemma 4 E4B (Gemma Team,

2026), Tiny-Aya Global (Salamanca et al., 2026), and Qwythos-9B (Empero AI, 2026)) and vary their inference configurations. LLM judges assess semantic quality and the requested language, while separate rules check measurable constraints such as length, required strings, and paragraph count. This distinction matters: Gemma has the highest holistic judge score, but Qwen passes hard constraints more often.

The provided sample has limited coverage of exact structural constraints. We therefore construct a diagnostic benchmark for grounded QA, constrained generation, and long-document summarization. We also build multilingual abstract-writing data and test QLoRA fine-tuning (Detrmers et al., 2023); the adapters reduce validation loss but not judged generation quality.

Our contributions are the following. (i) A configuration study of two finalist models, with paired bootstrap intervals and explicit multilingual analysis. (ii) An internal benchmark of 1,283 examples for context-grounded QA, exact output constraints, and 24-language abstract writing from full ACL papers, scored both semantically and with deterministic constraint checks. (iii) Negative results for supervised QLoRA and a preference-based correction stage, showing that lower validation loss need not improve output under the chosen judge. Code and data-construction scripts are available at <https://github.com/ampapacek/MIST>.

2 Task and Evaluation Setup

Task. MIST contains three subtasks. qa-context asks a question grounded in provided evidence, whereas qa-oeg asks for open-ended generation and often specifies exact structural or lexical constraints. sum-sum requires a summary in the requested language from source material that may be written in another language. The official test release covers 24 languages. It

*Disclosure of conflict: PS is also an organizer of MIST.

contains 12,775 prompts: 8,640 QA-context, 2,359 QA-OEG, and 1,776 summarization instances.

Development data. We use 10% of the 23,919 provided sample examples as our development set, retaining 80% for training and 10% as a held-out internal test set. We sample the splits stratified by task, target language, and source dataset to maintain balanced coverage. The resulting development set contains 2,392 examples: 1,491 QA examples and 901 summarization examples.¹ These examples primarily test answer correctness, summarization quality, and the requested output language, but provide limited coverage of the exact structural and lexical constraints found in the qa-oeg test prompts.

Internal evaluation benchmark. We therefore create an internal benchmark with three subsets aligned with the task. Appendix Table 11 gives abridged examples. The 755 qa-context items use fictional closed-world passages and include both answerable and unanswerable questions. GPT-5.6 Sol (OpenAI, 2026) generated English drafts, which we manually revised before translating passages, questions, and answers into several language combinations with DeepSeek V4 Pro (DeepSeek-AI, 2026). Each answer must be one sentence; unanswerable items require a fixed fallback in the answer language. The 288 qa-oeg items cover length, lexical, structural, and formatting constraints. The summarization subset contains 240 items: 10 full ACL papers sampled from those covered by the MCIF dataset (Papi et al., 2026), each summarized in all 24 target languages. Their references are translations of the English gold abstracts. These longer inputs expose context-window, completion, and length-control failures not common in the provided sample.

Metrics. We score standard development outputs with DeepSeek-v4-pro and repeat the finalist comparison with GPT-OSS-120B (OpenAI, 2025). The internal benchmark is evaluated only with GPT-OSS. Each judge sees the instruction, prediction, expected language, and reference when one exists. On a 0–10 integer scale, QA is scored for correctness, completeness, fluency, and language correctness; summarization uses faithfulness, coverage, conciseness, fluency, and language correct-

ness. The reported *overall* score is a separate holistic judgment, not an average of these dimensions. QA-OEG additionally supplies an item-specific semantic rubric because many responses can satisfy the same constrained prompt. The GPT-OSS judge template is given in Appendix E; every released judgment records its model and rubric version.

For the internal benchmark, we also compute a strict score by multiplying the GPT-OSS score by binary gates for normal completion, whether the judge identifies the response as being in the requested language, and all applicable hard constraints. Only the latter constraints are deterministic: word or character counts, required and forbidden strings, sentence, line and paragraph structure, exact endings, numbered lists, acrostics, and email structure. This distinction separates semantic quality, judge-assessed language choice, and rule-based instruction adherence.

All development-set intervals are 95% percentile bootstrap intervals obtained by resampling the evaluated examples 10,000 times with seed 3407. For a contrast between two configurations we resample the same examples for both systems and take the difference within each resample, so the interval describes paired per-example variation rather than the overlap of two independent means. Intervals are not corrected for multiple comparisons.

For each judge, configuration comparisons use the shared completed subset: examples with a normally finished prediction and a complete judgment in every compared configuration. This differs from the gated strict score used for the internal benchmark.

The official results considered here are from the automatic evaluation. QA-context is ranked by majority-vote verification accuracy from Command-A+, GLM-1, and Gemma-4-31B; QA-OEG by majority-vote rule pass rate from the same ensemble; and summarization by language-gated ROUGE-L F1, assigning zero to outputs that fail language identification. These metrics differ from our holistic development scores and deterministic internal gates.

3 Model, Prompt, and Hyperparameter Selection

3.1 Candidate models

We consider four open-weight models, whose specifications are summarized in Table 1.

Qwen and Gemma are our main candidates be-

¹The provided sample data cover a few target languages beyond the official 24, so our development split contains 27 target languages for QA and 24 for summarization.

Model	Params	Context	Langs
Tiny-Aya Global	3.35B	8K	70+
Gemma 4 E4B	4.5B eff.	128K	35+
Qwen 3.5 9B	9B	262K	201
Qwythos-9B	9B	1M	201 [†]

Table 1: Candidate models. Information is taken from the model reports or cards. Gemma 4 E4B is reported in effective parameters (about 8B with embeddings). [†]Qwythos-9B inherits the language coverage of its Qwen 3.5 9B base, but its additional reasoning-trace training is almost entirely English.

cause both fit the parameter limit and offer long context windows and broad multilingual support (Qwen Team, 2026; Gemma Team, 2026). Tiny-Aya represents a smaller, explicitly multilingual alternative (Salamanca et al., 2026). Qwythos is a reasoning-oriented Qwen fine-tune (Empero AI, 2026).

3.2 Selection

We first compare Gemma, Qwen, and Tiny-Aya on a balanced 457-example screen. Tiny-Aya obtains 5.39 on QA and 4.91 on summarization, compared with 7.45–7.47 and 6.68–6.74, respectively, for the best Qwen and Gemma configurations. Its 8K context window also cannot accommodate several longer inputs, the longest input has about 7,800 tokens, which leaves almost no room for an answer. We therefore drop Tiny-Aya from the larger development experiments.

For the two finalist models, we compare no system prompt (none) with a short task-oriented prompt (instruct), and reasoning on with reasoning off. For Gemma we additionally compare three decoding settings: the model card recommendation (temperature 1.0, top- p 0.95, top- k 64), a lower temperature of 0.5, and greedy decoding. Qwen uses its recommended settings, temperature 0.7 with top- p 0.8 and top- k 20 for reasoning off, and temperature 1.0 with top- p 0.95 and top- k 20 for reasoning on, both with a presence penalty of 1.0. In preliminary runs, greedy decoding without a presence penalty made Qwen loop and produce clearly more unfinished, repetitive answers, so we did not include a greedy Qwen configuration in the grid.

The complete instruct system prompt: “You are {assistant_role}. Follow all explicit task constraints. Use any explicitly requested output language; otherwise respond in {language}.” The role is “an assistant for question answering” for both

Configuration	All	QA	Sum
Gemma default	6.790	7.143	6.193
Gemma instruct/on	7.315	7.788	6.513
Qwen default	6.546	6.964	5.840
Qwen instruct/on	7.150	7.781	6.083

Table 2: Principal DeepSeek development results on 2,339 shared examples. Gemma uses greedy decoding. “Default” denotes no system prompt and reasoning off. Confidence intervals and the full grid are in Appendix B.

QA tasks and “a multilingual summarization assistant” for summarization; language is filled from the required answer-language field. The none condition adds no system message.

The standard development evaluation favors Gemma greedy with instruct system prompt and reasoning on. Under DeepSeek it scores 7.315 [7.195, 7.429] overall, compared with 7.150 [7.036, 7.266] for Qwen instruct/on; the paired difference is +0.165 and comes almost entirely from summarization (+0.430), while QA is indistinguishable (+0.007 [−0.127, +0.143]). GPT-OSS preserves the same overall ordering, but gives Qwen the highest QA score.

For these tested contrasts, the prompt and reasoning effects are larger than the difference between the two models. With reasoning on, adding the instruct prompt raises the score by +0.428 for Gemma greedy and +0.190 for Qwen. With the instruct prompt, enabling reasoning raises it by +0.280 for Gemma greedy and +0.350 for Qwen. Taken together, moving each model from its default condition (no system prompt, reasoning off) to its best configuration gains +0.53 for Gemma (6.790 to 7.315) and +0.60 for Qwen (6.546 to 7.150), that is, roughly three times the gap between the two models in their respective best settings.

The internal benchmark changes the practical conclusion: Qwen is substantially better on strict QA-OEG. We therefore select Qwen as the primary system and keep the higher-scoring Gemma as the secondary system.

Table 2 summarizes the principal development comparisons behind this selection. The dimension scores in Appendix Table 6 locate the summarization gap: Gemma is higher in faithfulness and coverage, while Qwen is more concise. For QA, Qwen is slightly more correct and complete, and Gemma more fluent. Requested-language scores are near the ceiling for both models. These values describe the judge’s decomposition and are not independent human measurements.

For the primary Qwen submission, we use no system prompt with reasoning on for QA-context, the instruct prompt with reasoning on for QA-OEG, and the instruct prompt with reasoning off for summarization. Its reasoning-on QA configurations use temperature 1.0, top- p 0.95, and top- k 20, while summarization uses temperature 0.7, top- p 0.8, and top- k 20; all three use a presence penalty of 1.0. The secondary Gemma submission uses the instruct prompt with reasoning on for both QA tasks and reasoning off for summarization, with greedy decoding and a presence penalty of 0.0 throughout. Both systems use a 24K model context and a 12K output budget.

Two of these choices do not follow the development ranking and are instead justified by the internal benchmark in Table 3. Qwen uses no system prompt for qa-context, because without it Qwen passes the deterministic checks more often (96.0% versus 93.9% of rows) and obtains a slightly higher judge score (8.828 versus 8.801), and qa-context is by far the largest test subtask. Both systems use reasoning off for sum-sum, because reasoning on lowers the long-paper summarization score for Qwen (8.013 to 7.746) and does not help Gemma (8.292 to 8.271), even though it helps on the shorter development summaries.

Reproducibility. All generation runs use vLLM with a fixed seed of 3407 and the decoding parameters listed above, on NVIDIA L40S (48 GB) GPUs provided by MetaCentrum. Fine-tuning uses 4-bit QLoRA through Unsloth on the same hardware. Every generated record retains both the raw output and the final answer after removal of thought-channel markup, together with the finish and stop reasons, output-token counts, and run metadata.

4 Training Data Creation

Summarization is weaker than QA for both finalists, so our training data target abstract writing. We use 500 ACL Anthology papers: 100 MCIF scientific-talk documents from the provided sample and 400 additional pre-2025 papers.

For the 100 MCIF papers, the input is the paper body from the Introduction to the References, with the abstract removed, and the target is the corresponding abstract. The original data provide gold abstracts in four languages—English, German, Italian, and Simplified Chinese—and we create silver translations for the remaining 20 MIST languages, giving 2,400 paper-language pairs of

which 1,920 fall in the training split. We use paper-level 80/10/10 splits so that the same papers do not occur in both training and evaluation. For the other 400 papers, we retain each official English abstract and translate it into the other 23 languages.

We train separate variants on final-answer-only examples and on examples augmented with synthetic reasoning traces from DeepSeek V4 Pro. Most non-English targets are silver and may contain systematic translation artifacts.

5 Fine-Tuning Experiments (Negative Result)

We use 4-bit QLoRA with completion-only loss and train adapters on the MCIF and ACL abstract-writing data. We evaluate checkpoints against the matching base model on identical inputs and with identical decoding parameters, so that only the adapter differs.

For Qwen on the 9,600-example mixture, token-normalized training loss decreases from 1.512 to 1.170 and pooled validation loss from 1.573 to 1.189. GPT-OSS, however, scores the base model at 8.080, checkpoint 97 at 6.626, and checkpoint 291 at 6.957. Inspection of low-scoring outputs finds clarification requests, language switches, unrelated-paper hallucinations, and repetition. Thus the lower completion loss does not improve quality under this evaluation.

Gemma adapters trained on the 24-language MCIF data also score below their base models (−3.61 for E2B and −3.34 for E4B). Lower LoRA rank and learning rate, one epoch, earlier checkpoints, and updating only the final 14 layers do not reverse the result. Possible explanations include multilingual interference, silver-target artifacts, and mismatch between reference style and the judge; our experiments do not distinguish among them.

A run on the provided sample data produces shorter outputs closer to reference length, but the judge still prefers the base model and assigns the adapter more hallucination, wrong-language, and low-coverage failures. This contrast makes the result metric-dependent: reference-length agreement improves while the judge score declines.

An English-only Gemma 4 E2B pilot has the same direction: the base model scores 7.77 and the best adapter 7.60 on 30 papers (4 wins, 17 ties, and 9 losses).

Finally, we apply Direct Preference Optimization (DPO) to 1,917 pairs of Gemma 4 E4B drafts

System	Prompt/reas.	Diag.	Pass (%)	MCIF
Qwen	none / off	7.568	74.3	6.988
Qwen	none / on	8.466	88.6	7.638
Qwen	instruct / off	7.627	70.1	8.013
Qwen	instruct / on	8.467	86.7	7.746
Gemma	none / off	8.223	73.3	6.083
Gemma	none / on	8.424	79.0	7.188
Gemma	instruct / off	8.265	77.5	8.292
Gemma	instruct / on	8.456	82.2	8.271
Qwythos	instruct / on	7.819	77.7	6.992

Table 3: Internal benchmark under GPT-OSS-120B. *Diag.* is the mean score over the 1,043 diagnostic examples (755 QA-context and 288 QA-OEG), *Pass* is the percentage of those examples passing every applicable deterministic check, and *MCIF* is the mean score on the 240 long-paper summarization examples.

and minimal DeepSeek corrections toward the references (Rafailov et al., 2023). The best checkpoint is 0.02 below the base score of 6.91, and longer training increases the deficit to -0.16 . The corrected targets inherit the concise reference style, which may be mismatched with the judge, but reference scoring is needed to test this explanation.

Overall, none of the training variants beat the corresponding base model under the judge, so our submitted systems use the base models.

6 Results and Analysis

Table 2 gives the principal DeepSeek comparison, and Appendix Table 5 repeats the finalist comparison with GPT-OSS. Both judges rank Gemma greedy instruct/on first overall. Qwen is effectively tied with Gemma on QA under DeepSeek and leads QA under GPT-OSS, whereas Gemma leads summarization. The full DeepSeek grid is in Appendix B.

The Gemma temperature ablation is the one place where the ranking does not resolve. Greedy decoding and temperature 0.5 are statistically tied in the finalist setting ($+0.054$ [-0.012 , $+0.121$]), and greedy decoding produces more unfinished generations (29 against 14 across the four prompt and reasoning settings). We use greedy decoding because it is deterministic and has the higher point estimate.

The internal benchmark explains why we do not simply submit the model with the highest overall mean. Under strict gating, Qwen instruct/on scores 4.931 on QA-OEG, compared with 3.122 for Gemma greedy instruct/on. Gemma’s raw QA-OEG quality is higher (7.829), but it frequently violates exact word counts, required strings, para-

graph counts, and other structural constraints. On QA-context, the selected systems are much closer: their strict language-macro scores are 8.410 for Qwen and 8.426 for Gemma. Table 3 shows the same effect at the level of the whole diagnostic set: the two systems are within 0.01 of each other on semantic score in their best configuration, but Qwen passes the deterministic checks in 86.7% of rows against 82.2% for Gemma.

The long-paper summarization result differs from QA-OEG. Gemma writes longer answers than Qwen (median 1,506 against 944 characters on development summarization). In QA-OEG, where length and structure are checked, Qwen has the stronger strict score. On the MCIF subset, where we report only the judge score, Gemma leads with 8.292 in its greedy instruct/off configuration, ahead of Qwen’s 8.013.² Qwen has no length stops in this 240-example condition, compared with one for Gemma instruct/off. We do not use length matching to favor Qwen: Gemma places more outputs within $\pm 20\%$ of the reference character length (152 versus 121). These comparisons do not establish that output length itself causes the score difference. Instead, Qwen’s stronger strict QA-OEG score motivates it as the safer primary system under exact instructions, while Gemma is a strong secondary system for summarization.

Multilingual behavior. Appendix Table 8 directly contrasts English with the macro-average over the other development languages. Qwen is slightly ahead in English for both tasks. Outside English, Gemma is ahead, especially on summarization. Thus the aggregate result is not simply an English ranking reproduced across languages. The detailed tables in Appendix C show further variation: Qwen leads QA in 15 of 27 languages, whereas Gemma leads summarization in 21 of 24. On the strict internal benchmark, the task-specific Qwen configurations lead QA-OEG in 20 of 24 languages, while Gemma leads QA-context in 14 of 24. These are descriptive comparisons because some language slices contain only 5–10 examples.

Official automatic results. Table 4 gives the primary scores and ranks for our two submissions, as reported in the shared-task findings (Schmidtová et al., 2026).

²A summarization-focused prompt variant moves Qwen to 8.063 and Gemma to 8.065; we did not adopt it for the submission.

System	Overall	QA-C	OEG	Sum
Gemma (B)	1 / 1.67	2 / 80.93	1 / 88.45	2 / 17.41
Qwen (P)	2 / 2.67	1 / 81.63	2 / 82.25	5 / 14.71

Table 4: Official automatic results for Gemma variant B (B) and Qwen primary (P), shown as rank / score. The Overall column gives rank / mean rank. QA-C and OEG use their LLM-ensemble primary metrics; summarization uses language-gated ROUGE-L F1.

Our systems occupy the top two overall positions in the official automatic evaluation. Qwen wins QA-context, while Gemma wins QA-OEG and ranks second in summarization, only 0.14 points behind the task winner. The QA-OEG result does not reproduce Qwen’s advantage on our internal deterministic checks: Gemma scores 88.45% and Qwen 82.25%. The evaluations are not equivalent, because the official primary metric aggregates LLM-judge decisions about individual rules, whereas our strict score applies programmatic gates. The result therefore shows that our diagnostic benchmark did not predict the official ordering for this subtask.

Conclusion

Under our judges, the tested inference settings matter more than the difference between the finalists. On the 0–10 judge scale, moving from no system prompt and reasoning off to the best configuration raises Gemma from 6.790 to 7.315 and Qwen from 6.546 to 7.150. The short system prompt and model-native reasoning improve the aggregate development scores, although the internal benchmark leads us to disable one or both for some submitted subtasks.

Gemma 4 E4B gives the strongest overall development score, especially outside English summarization, whereas Qwen 3.5 9B follows exact multilingual constraints more reliably on our internal benchmark. Neither QLoRA on multilingual abstracts nor preference optimization improves our judge score, so both submissions use the base models.

In the official automatic evaluation, the Gemma secondary system ranks first overall and the Qwen primary system second. Qwen wins QA-context, while Gemma wins QA-OEG and places second in summarization. Thus, the two systems occupy the top two overall positions, although the official QA-OEG ordering differs from our internal deterministic evaluation.

Limitations

LLM-based evaluation. Our conclusions depend on LLM judges and lack human validation. Although both judges give the same overall finalist ordering, the internal benchmark uses only GPT-OSS and one rubric. Language correctness is not checked independently, and without a human or unrestricted-model topline, the absolute remaining quality gap is not calibrated.

Fine-tuning conclusions. Fine-tuning may be penalized by a mismatch between concise references and judge preferences. Since we have not scored the gold references with the same rubrics, conclusions about adapter degradation and this mismatch remain conditional on the judge.

Statistical uncertainty. Language slices contain only about 25–30 QA-context examples and 10 QA-OEG and summarization examples, and we use a single generation seed, so differences are descriptive. The bootstrap intervals cover example-sampling uncertainty only, not generation or judge variation, and are not corrected for multiple comparisons.

Silver data and possible contamination. Outside the four MCIF gold languages, the training targets and MCIF evaluation references are machine translations, so shared artifacts may affect both. The public pre-2025 ACL papers may also have appeared in the candidate models’ pre-training data.

Acknowledgments

This work has been supported by the project TQ12000040 (CZDEMOS4AI) financed by the Technology Agency of the Czech Republic (www.tacr.cz) within the Sigma 3 Program. It was also supported by the Ministry of Education, Youth and Sports of the Czech Republic through the e-INFRA CZ research infrastructure (ID:90254). We gratefully acknowledge MetaCentrum and CERIT-SC for providing the computing infrastructure and services we used to evaluate the open-weight LLMs in our experiments.

It was also supported by Charles University projects GAUK 252986 and SVV 260 698 and co-funded by the European Union (ERC, NG-NLG, 101039303).

Patrícia Schmidtová is one of the MIST organizers. She took part in this work only as a supervisor and mentor and did not share any non-public task information.

References

- DeepSeek-AI. 2026. [DeepSeek-V4-Pro model documentation](#).
- Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. 2023. QLoRA: Efficient finetuning of quantized LLMs. In *Advances in Neural Information Processing Systems*, volume 36.
- Empero AI. 2026. [Qwythos-9B model card](#).
- Gemma Team. 2026. [Gemma 4 technical report](#). Technical report.
- Tom Kocmi, Sweta Agrawal, 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 Schmidová, 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.
- OpenAI. 2025. [gpt-oss-120b and gpt-oss-20b model card](#).
- OpenAI. 2026. [GPT-5.6 Sol model documentation](#).
- Sara Papi, Maike Züfle, Marco Gaido, Beatrice Savoldi, Danni Liu, Ioannis Douros, Luisa Bentivogli, and Jan Niehues. 2026. [MCIF: Multimodal crosslingual instruction-following benchmark from scientific talks](#). In *The Fourteenth International Conference on Learning Representations*.
- Qwen Team. 2026. [Qwen3.5-9B model card](#).
- 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. In *Advances in Neural Information Processing Systems*, volume 36.
- Alejandro R. Salamanca et al. 2026. [Tiny Aya: Bridging scale and multilingual depth](#). *arXiv preprint arXiv:2603.11510*.
- Patricia Schmidová, Katia Artemova, Seth Aycok, Niyati Bafna, Tom Kocmi, Philipp Koehn, Danni Liu, Nam Luu, Sara Papi, Beatrice Savoldi, Mariya Shmatova, Vilém Zouhar, Maike Züfle, and Pinzhen Chen. 2026. Findings of the WMT26 multilingual instruction shared task: Small models are not yet polyglots. In *Proceedings of the Eleventh Conference on Machine Translation*. Association for Computational Linguistics.

A Additional development results

Table 5 reports the finalist comparison under the secondary GPT-OSS-120B judge, while Table 6 breaks down the DeepSeek scores by evaluation dimension.

Configuration	All	QA	Sum
Gemma greedy, instruct/on	7.746	7.862	7.549
Qwen, instruct/on	7.654	7.974	7.113
Qwythos, instruct/on	7.318	7.648	6.760

Table 5: Development means under the secondary GPT-OSS-120B judge on its own shared completed subset of 2,289 examples. The overall ordering matches Table 7, but Qwen is best on QA.

Task	Dimension	Gemma	Qwen
QA	correctness	7.931	7.980
	completeness	7.903	8.082
	fluency	9.561	9.207
	language	9.976	9.957
Sum	faithfulness	7.399	6.656
	coverage	7.139	6.336
	conciseness	6.685	7.417
	fluency	8.888	8.300
	language	9.994	9.945

Table 6: Mean DeepSeek dimension scores for the finalist instruct/on configurations on the shared completed subset.

B Full development grid

Table 7 expands the principal comparison in Table 2 to all evaluated prompt, reasoning, and decoding configurations.

C Per-language development results

Tables 9 and 10 give the per-language means behind the aggregate scores in Table 7. All rows use the instruct prompt with reasoning on, and all scores come from the DeepSeek V4 Pro judge on the shared completed subset. Sample sizes are small and uneven, so the last column is descriptive and is not tested for significance.

D System prompt

This section gives the exact instruct template used in the development configurations reported in Table 7, followed by a representative QA-OEG item and its checks.

System prompt (instruct). The exact system-prompt template is:

You are {assistant_role}. Follow all explicit task constraints. Use any explicitly requested output language; otherwise respond in {language}.

For QA tasks, including QA-OEG, the assistant role is “an assistant for question answering”; for summarization, it is “a multilingual summarization assistant”. For QA, the target language is taken from the given question_lang field and inserted

Model	Prompt	Reasoning	All	QA	Summarization	Unfin.
Gemma $t=1.0$	none	off	6.655 [6.525, 6.781]	7.104 [6.918, 7.292]	5.894 [5.763, 6.023]	1
Gemma $t=1.0$	none	on	6.759 [6.635, 6.885]	7.335 [7.155, 7.512]	5.785 [5.665, 5.902]	1
Gemma $t=1.0$	instruct	off	6.883 [6.755, 7.008]	7.269 [7.079, 7.447]	6.229 [6.105, 6.349]	1
Gemma $t=1.0$	instruct	on	7.147 [7.029, 7.265]	7.737 [7.562, 7.905]	6.148 [6.037, 6.261]	0
Gemma $t=0.5$	none	off	6.732 [6.607, 6.860]	7.074 [6.879, 7.259]	6.153 [6.029, 6.274]	9
Gemma $t=0.5$	none	on	6.892 [6.768, 7.016]	7.433 [7.254, 7.610]	5.977 [5.858, 6.093]	1
Gemma $t=0.5$	instruct	off	7.038 [6.913, 7.162]	7.431 [7.248, 7.612]	6.374 [6.250, 6.496]	3
Gemma $t=0.5$	instruct	on	7.261 [7.141, 7.378]	7.748 [7.578, 7.912]	6.436 [6.322, 6.548]	1
Gemma greedy	none	off	6.790 [6.663, 6.917]	7.143 [6.959, 7.331]	6.193 [6.067, 6.316]	8
Gemma greedy	none	on	6.886 [6.759, 7.014]	7.405 [7.224, 7.583]	6.008 [5.886, 6.122]	6
Gemma greedy	instruct	off	7.035 [6.908, 7.160]	7.405 [7.220, 7.586]	6.409 [6.288, 6.527]	11
Gemma greedy	instruct	on	7.315 [7.195, 7.429]	7.788 [7.621, 7.961]	6.513 [6.406, 6.621]	4
Qwen	none	off	6.546 [6.422, 6.671]	6.964 [6.788, 7.137]	5.840 [5.704, 5.976]	8
Qwen	none	on	6.960 [6.842, 7.079]	7.499 [7.328, 7.669]	6.048 [5.932, 6.161]	17
Qwen	instruct	off	6.800 [6.674, 6.925]	7.356 [7.184, 7.535]	5.860 [5.718, 5.995]	10
Qwen	instruct	on	7.150 [7.036, 7.266]	7.781 [7.614, 7.944]	6.083 [5.974, 6.189]	9

Table 7: Full development grid under the DeepSeek V4 Pro judge: mean holistic score with a 95% bootstrap interval on the shared completed subset of 2,339 of the 2,392 development examples (1,470 QA and 869 summarization). *Unfin.* counts generations that did not finish normally, over all 2,392 attempted examples. Bold marks the best configuration in each column.

Task	Slice	Gemma	Qwen
QA	English	8.280	8.460
QA	Other languages	7.685	7.576
Sum	English	6.940	6.960
Sum	Other languages	6.387	5.882

Table 8: English score and unweighted language-macro score for all other languages under DeepSeek. Both systems use instruct/on; Gemma is greedy.

into the language placeholder. Thus, for the English QA-OEG example below, the rendered system prompt is:

You are an assistant for question answering. Follow all explicit task constraints. Use any explicitly requested output language; otherwise respond in English.

QA-OEG task example. One English QA-OEG item asks:

Write a 120-to-150-word proposal for a fictional neighborhood seed exchange. Use exactly three paragraphs and no bullet points. The first paragraph must identify the problem, the second must describe the exchange, and the third must name one expected benefit. Include the exact expressions “shared shelf” and “monthly ledger”. Do not use the word “free”.

QA-OEG rubric and deterministic checks. Its item-specific semantic rubric requires that the proposal explain a workable neighborhood seed exchange and that each paragraph fulfill its assigned role. The associated deterministic checks enforce the 120–150-word range, exactly three paragraphs, no bullet points, both required expressions, and exclusion of the forbidden word.

E Judge prompt

The GPT-OSS development and internal-benchmark runs use the following system instruction:

You are a demanding, calibrated evaluator for a multilingual instruction-following shared task. You assess a model’s OUTPUT against the user’s INSTRUCTION. Be critical and discriminating: top scores (9–10) are reserved for output with no discernible flaw; reserve 5–6 for output with real but non-fatal problems, and do not anchor on high scores. Judge only what is asked; never reward verbosity, padding, or merely restating the instruction. A single unsupported claim, factual error, or missing key point should visibly lower the relevant score. Always write the rationale in English, regardless of the output’s language. Respond with a single JSON object and nothing else.

The user message contains labeled fields for task, expected language, instruction, optional reference, and model output. It asks for integer scores from 0 to 10, where 9–10 is near-flawless, 7–8 good with minor issues, 5–6 acceptable with real problems, 3–4 poor, and 0–2 unusable or wrong. The task criteria are those listed in Section 2. The final instruction is:

Score overall directly as holistic task quality; it is not the average of the dimensions. Return only a JSON object containing every task dimension, overall, detected_language, target_language_match, and a one-sentence English rationale.

F Internal benchmark examples

Table 11 illustrates the inputs, targets, and checks for the three internal benchmark subsets described in the main text.

Language	$G_{0.5}$	G_0	Qwen	n
Yoruba	1.30	1.15	1.96	27
Czech	4.40	5.40	3.80	5
Marathi	6.58	7.03	5.56	59
Turkish	6.71	6.76	6.47	59
Hindi	8.00	7.46	6.83	65
Italian	7.17	7.44	7.68	81
Swahili	7.70	7.23	7.70	30
Thai	8.07	7.37	7.33	30
Japanese	7.62	7.59	7.66	90
Spanish	7.63	7.47	7.82	60
Vietnamese	7.64	7.73	7.59	59
Russian	7.33	7.77	7.92	92
Arabic	7.55	7.74	7.81	94
Korean	7.93	7.91	7.99	90
Chinese (Simpl.)	7.62	7.94	8.42	86
Indonesian	7.91	7.89	8.37	95
Portuguese	8.12	8.14	7.97	59
Bengali	8.69	8.38	7.50	16
C. Kurdish (Sorani)	8.15	8.41	8.04	27
English	8.12	8.28	8.46	57
German	8.48	8.53	8.36	81
French	8.47	8.50	8.59	58
Finnish	9.00	9.07	9.27	30
Haitian Creole	9.07	9.07	9.33	30
Slovak	9.10	9.23	9.17	30
Persian	9.47	9.07	9.27	30
Telugu	9.93	9.53	8.57	30

Table 9: QA by language, sorted by the lowest mean across the three systems. $G_{0.5}$ and G_0 are Gemma at temperature 0.5 and greedy.

Language	$G_{0.5}$	G_0	Qwen	n
Yoruba	3.83	3.83	3.33	6
Swahili	5.19	5.65	3.81	31
Marathi	5.79	5.86	4.48	42
Bengali	5.46	5.86	5.49	35
Telugu	5.81	6.28	4.94	32
Slovak	5.22	6.11	5.89	9
Thai	6.00	5.94	5.72	32
Haitian Creole	6.40	6.70	5.40	10
Japanese	6.31	6.27	5.93	45
Persian	6.16	6.46	6.08	37
Arabic	6.32	6.51	6.02	47
Russian	6.26	6.66	6.02	47
Hindi	6.40	6.60	5.94	47
Turkish	6.59	6.65	6.15	46
Korean	6.60	6.56	6.28	43
Indonesian	6.66	6.40	6.47	47
French	6.77	6.68	6.66	44
Spanish	6.96	6.72	6.57	47
Italian	7.00	6.95	6.47	19
Portuguese	6.89	6.91	6.81	47
Vietnamese	7.15	6.91	6.57	46
English	6.83	6.94	6.96	48
Chinese (Simpl.)	7.00	6.98	7.26	42
German	7.45	7.40	7.00	20

Table 10: Summarization by language, sorted by the lowest mean across the three systems.

Subset	Abridged input	Target / checks
QA-context	A French passage states that fictional Selnor thread charts record shared journeys; a Bengali question asks what they record.	One Bengali sentence: “Selnor thread charts record shared journeys.” For unanswerable questions, output only the localized fallback.
QA-OEG	Explain first-come-first-served processing using only a bakery analogy, in 100–130 Arabic words, without five forbidden technical terms.	A coherent Arabic analogy; word range and forbidden terms are checked programmatically.
Summarization	Write an approximately 200-word Arabic abstract of a full ACL paper with the original abstract removed.	A formal third-person Arabic abstract; semantic scoring uses the translated gold abstract.

Table 11: Abridged examples from the three internal benchmark subsets. Full records, references, rubrics, and constraint specifications are released with the benchmark.