SentinelQA: Task-Level Routing and Output Verification for Multilingual Instruction Following

Asher Egerton-Idehen
Independent Researcher
asheregerton1@gmail.com

Abstract

We describe team Sentinel's submission to the WMT26 MIST shared task on multilingual instruction following. We allocate the budget across a pipeline built on `gemma-3-4b-it`: a fixed task-level router that selects between fine-tuned and base behaviour, and a deterministic verification layer checking language, length, repetition and chat-template artefacts. Our submitted system uses **9.46B** parameters. A proposed shared-base LoRA configuration would use 5.16B, but was not evaluated end-to-end. Fine-tuning raises context-based QA token-F_1 from 0.267 to 0.74. The verification layer alters roughly 9% of outputs and raises our reference-free compliance measure by 2.2 points; because a replacement is accepted only when it passes more of the same checks, we report this as the yield of the intervention rather than an identified effect. Yoruba compliance falls from 0.816 under base to 0.675 under the augmented full fine-tune, recovering partially to 0.754 with LoRA; these single-seed observations do not isolate augmentation or tuning method as the cause.

1 Introduction

The WMT26 MIST shared task evaluates multilingual instruction following across three subtasks: context-based question answering (`qa-context`), open-ended generation QA (`qa-oeg`), and mono- and cross-lingual summarization (`sum-sum`), in 24 languages and subject to a limit of 10B parameters across all deployed components.

The test set contains 12,775 items: 8,640 `qa-context` (67.6%), 2,359 `qa-oeg` (18.5%) and 1,776 `sum-sum` (13.9%). Most items are cross-lingual. Computed from the item identifiers, 95.8% of `sum-sum` items and 96.1% of `qa-context` items ask for output in a language other than that of the source document, almost always English; `qa-oeg` is entirely monolingual.

Our submission is built on `gemma-3-4b-it` (Gemma Team, Google DeepMind, 2025), a 4.37B-parameter instruction-tuned model. We submit three systems. The primary system routes each item by subtask: fine-tuned behaviour for `qa-context`, base behaviour for `qa-oeg` and `sum-sum`. The two contrastive systems are the fine-tuned model alone and the base model alone, the latter doubling as our untuned baseline. All three pass every generation through the same verification layer (§4); Figure 1 shows the proposed shared-base serving configuration.

2 Related Work

The verification layer is a minimal instance of the pattern studied in the self-correction literature, e.g. Self-Refine (Madaan et al., 2023) and constraint decomposition approaches such as DECRIM (Ferraz et al., 2024), but its checks are external and non-neural rather than model-judged. Huang et al. (2024) report that LLMs struggle most with autonomous error *detection*, which is the step we delegate to fastText classifiers (Joulin et al., 2017, 2016) trained for language identification (Kargaran et al., 2023; Burchell et al., 2023) and to deterministic parsers. In machine translation, reranking an n-best list with a quality-estimation model (Rei et al., 2020; Fernandes et al., 2022; Freitag et al., 2022) is the established way to recover from a bad sample; our single-candidate retry is a cheaper point on that trade-off. Following the WMT25 MIST findings (Kocmi et al., 2025) we treat chrF++ (Popović, 2017) as reportable but not decisive, given its weak correlation with human judgment on low-resource languages.

3 Data

We train on the sample release provided by the shared-task organizers (Chen and WMT26 MIST Organizers, 2026), whose per-language table sums

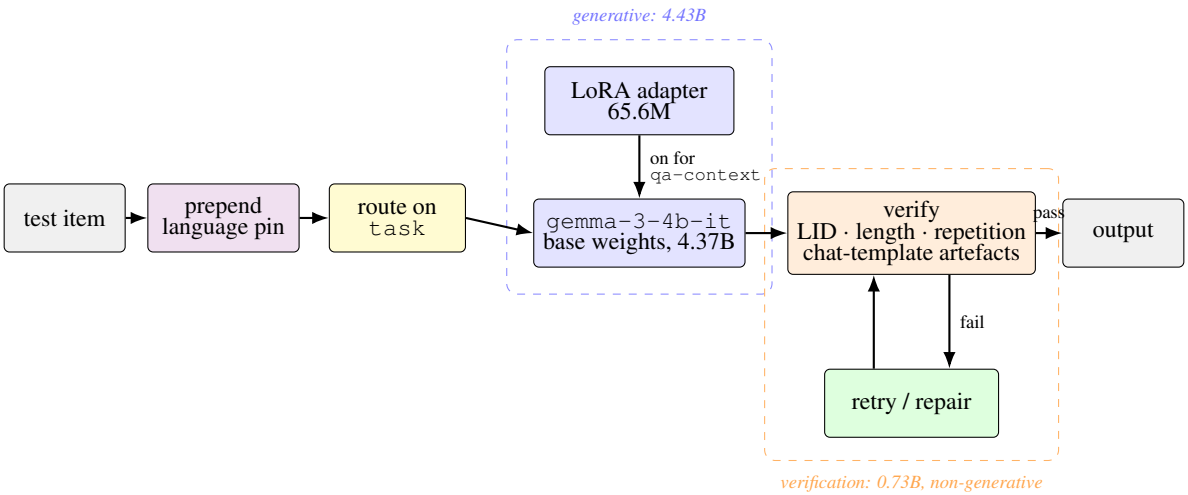

Figure 1: Proposed shared-base configuration of SentinelQA (5.16B parameters; not evaluated end-to-end). The fixed task lookup enables the adapter for `qa-context` and disables it for the other subtasks. The submitted system instead used a separate full fine-tune alongside base (9.46B). Both designs use a target-language pin and four verification checks, with conditional retry or repair (§4).

to the 23,919 rows we audited.

Its sources are TyDi QA (Clark et al., 2020), BELEBELE (Bandarkar et al., 2024), MCIF (Papi et al., 2025), the Aya dataset (Singh et al., 2024), CrossSum (Bhattacharjee et al., 2023), WikiLingua (Ladhak et al., 2020) and the WMT25 MIST OEG submissions.

We first froze an evaluation split, sampling $\min(100, \lceil 0.15n \rceil)$ items from each (language, subtask) cell into a held-out set of 3,599 rows, then made it read-only and hash-pinned. For cells with no data at all — Yoruba `qa-context`, for instance — we filled with teacher distillation from `Qwen3-235B-A22B-Instruct-2507` for `qa-oeg` (2,043 rows), NLLB-200-3.3B (NLLB Team et al., 2022) translation into six languages (747 rows) and native BELEBELE `qa-context` for Bengali, Czech and Yoruba (264 rows). Augmented rows passed the same language and degeneracy checks used at inference. The final corpus is 23,117 rows.

Two properties of the data bear on the numbers below. BELEBELE is parallel, so one passage appears in every language under a shared identifier; we recovered identifiers for all 847 held-out BELEBELE rows and excluded from training every row carrying one of the resulting 400 identifiers, which removes 84% of otherwise-usable BELEBELE but keeps held-out passages out of training. Separately, 48 held-out rows carry an English reference despite a non-English target, all from the three languages outside WikiLingua's 18-language coverage; we score them as a separate slice, so held-out results below use the remaining n=3,551.

4 System

Routing. The router is a fixed lookup on the `task` field with no learned component: `qa-context` takes fine-tuned behaviour, `qa-oeg` and `sum-sum` take base behaviour. Items are copied verbatim from the corresponding stream, so the routed submission is a deterministic recombination of two systems we also submit separately.

The three subtasks rest on evidence of different strength. `qa-context` (67.6% of the test set) is decided by a large margin: token-F_1 rises from 0.267 to 0.74 (§6.1). `sum-sum` (13.9%) is decided by length: the fine-tuned models produce a median of 92–134 words on prompts requesting roughly 200, where base produces 210. For `qa-oeg` (18.5%), post-verification compliance differs by only 0.0025 (0.9806 base against 0.9781 fine-tuned), and we have no reference-based accuracy measure. Recorded test-generation diagnostics also favor base (425 flagged outputs versus 562 for Run B), but do not establish better answer quality. We routed it to base partly by analogy with `sum-sum`; this remains the weakest decision. The evidence included a 288-item test-prompt probe and test-output diagnostics, not solely held-out results, and does not establish routing generalization to a new test distribution.

Verification. Every generation is prefixed with a target-language system pin (*"Respond only in ⟨endonym⟩"*) and checked on four axes.

Language uses dual fastText identification with GlotLID v3 and OpenLID v3, returning PASS/FAIL/UNSCORABLE and rejecting text only when *both* models place it outside the accept set, with short strings marked unscorable rather than failed. **Length** uses a deterministic multilingual parser for explicit word- and sentence-count constraints with script-aware counting; it fires on 85.6% of test prompts, covering 100% of `qa-context` and `sum-sum` prompts and 22.1% of `qa-oeg` prompts. **Degeneracy** uses n-gram repetition detection. **Artefacts** checks for chat-template markers and leaked language-pin text. Parser and detector details are in Appendix A.

We summarise these outcomes as **compliance**. The retained held-out tables pool check rates, whereas the test analysis and bootstrap average per-item scores; Appendix A defines these distinct aggregations. Compliance is reference-free, so it can be computed on the test set, and it measures whether an output is well-formed rather than whether it is correct.

A failed check triggers a temperature-0 retry with a fixed correction appended; outputs still degenerate afterwards are regenerated with a repetition penalty, and under-length `sum-sum` drafts get one edit-based expansion pass. In every case the candidate replaces the original only if it passes strictly more checks, or fixes the language without breaking another axis.

Parameter budget. The submitted system holds base and a separate full fine-tune, totaling 9.46B with language identification (Table 1, left). A proposed alternative enables Run C's 65.6M adapter against shared base weights for `qa-context` and disables it otherwise, totaling **5.16B** (right). This is a calculated 45% parameter reduction, not a measured memory or speed gain. Run C was evaluated with merged checkpoints; neither the unmerged serving configuration nor a routed Run C submission was evaluated end-to-end. The available comparisons motivate LoRA as a candidate for future deployment, but do not establish equivalence to the submitted Run B system or identical generated outputs. A deployment recommendation requires serving validation and an end-to-end evaluation.

We count the fastText models throughout, since they run at inference and are not negligible at 0.73B; under the adapter accounting they are the second-largest component after the base model itself.

component	as submitted	proposed
`gemma-3-4b-it` (base)	4,365,656,432	4,365,656,432
second copy, merged FT	4,365,656,432	—
LoRA adapter	—	65,576,960
GlotLID v3	418,934,528	418,934,528
OpenLID v3	307,181,056	307,181,056
total parameters	**9,457,428,448**	**5,157,348,976**

Table 1: Parameter accounting against the task's 10B limit. The left column describes the submitted configuration; the right is a proposed shared-base LoRA configuration, not an evaluated submission.

5 Experimental Setup

All training used a single NVIDIA A100-SXM4-80GB with `bfloat16`, sequence length 4096 with packing, loss on assistant tokens only, effective batch size 64, three epochs, a cosine schedule with 3% warmup and seed 20260801. We trained three models (Table 2): Run A on the organizer sample only, Run B on the full augmented corpus, and Run C on the full corpus with LoRA (Hu et al., 2022) (r=32, α=64, all attention and MLP projections, 65.6M trainable parameters, 1.50% of the model). Run C's adapters are merged into the base weights at each checkpoint save for evaluation convenience, so all three runs are interchangeable under the same inference code; §4 discusses the proposed unmerged serving configuration. Inference used vLLM (Kwon et al., 2023), with first-pass temperatures 0.0 for `qa-context`, 0.3 for `sum-sum` and 0.7 for `qa-oeg`. Runs B and C both start from base, but differ in learning rate as well as tuning method: full fine-tuning uses `paged_adamw_8bit`, while LoRA uses `adamw_torch_fused` and adapter dropout 0.05. Their comparison therefore does not isolate the number of trainable parameters.

Checkpoints were selected by a rule fixed before any results existed: compliance plus `qa-context` F_1 on the native held-out slice, subject to an English/German retention gate, with chrF++ reported but not decisive (computed with sacreBLEU, Post, 2018). Both full fine-tunes peaked at epoch 2.

Comparisons use paired bootstrap resampling (Koehn, 2004) with B=1000; the finest p-value this resolves is $1/B$.

	Run A	Run B	Run C
data	sample	full	full
rows	20,063	23,117	23,117
method	full FT	full FT	LoRA
learning rate	1e-5	1e-5	1e-4
steps	141	159	159
final train loss	—	1.759	1.784

Table 2: The three training recipes, differing in data selection and optimization settings (§5). Run A's final loss was not recorded.

system	compl.	F_1	chrF++	agg.
base	0.8073	0.2671	25.09	1.0744
Run A (e2)	**0.8266**	0.7462	45.17	**1.5728**
Run B (e2)	0.8238	0.7405	44.98	1.5643
Run C (e3)	0.8154	**0.7499**	**45.39**	1.5653

Table 3: Held-out results for the selected checkpoint of each run; compliance pools check rates (Appendix A). "agg." is the unweighted sum of compliance and `qa-context` F_1; it has no independent interpretation and is reported only because it was the pre-registered selection rule.

6 Results

Held-out numbers use the native slice (n=3,551; F_1 over the n=1,454 `qa-context` items).

6.1 Held-out results

Table 3 gives the selected checkpoint of each run against the base model. Fine-tuning raises `qa-context` token-F_1 from 0.267 to roughly 0.74 across all three runs. Since the gold spans have a median of four words while the base model emits a median of 30 on the same items, much of this margin reflects acquisition of the expected output format rather than improved comprehension. Compliance varies much less; Run B is slightly ahead of base under the pooled-check aggregation. Appendix B gives a per-language breakdown.

Against the base model, Run B gains $+0.4743$ F_1 (95% CI $[+0.4521, +0.4976]$, $p < 1/B$).[1] The available paired comparisons among fine-tuned checkpoints do not resolve differences in compliance or F_1 on this single-seed evaluation; this is not an equivalence test. Their compliance intervals use per-item scores, not Table 3's pooled-check score, and therefore do not quantify uncertainty for that table's compliance gaps.

[1] Computed on Run B's epoch-3 checkpoint; the submitted epoch-2 checkpoint differs from base by $+0.4734$ on the same measure.

Yoruba regression and partial recovery. Yoruba compliance falls from 0.816 under base to 0.675 under Run B and recovers to 0.754 under Run C (Appendix B). Repetition flags rise from 23/78 to 55/78, then fall to 44/78. B-to-C recovery is concentrated in summaries (25 to 12 repetition flags), while open-ended QA flags increase from 30 to 32; recovery is not uniform across tasks. This slice contains 45 `qa-oeg` and 33 `sum-sum` items, with no `qa-context` items, whereas all 88 added Yoruba examples are `qa-context`. It therefore tests transfer to other tasks, not performance on the augmented task. Repeated article targets in the training corpus plausibly encourage repetitive generation, but the available evidence does not identify a causal mechanism. Base-to-B also introduces fine-tuning, and B-to-C changes the optimization recipe; neither contrast isolates augmentation or LoRA. A row audit additionally found two Yoruba train/held-out input duplicates after normalization and three held-out prompts with falsely parsed length constraints. We retain the historical scores and treat the regression as descriptive, pending clean evaluation and per-language uncertainty estimates.

6.2 Verification layer

The retained aggregate analysis compares the first cleaned generation, before retry or repair, with the shipped text on the full test set (Table 4). The three single-model systems alter 8.6–9.9% of outputs and add 2.4–2.8 compliance points; the routed system, whose items are copied from those streams, adds 2.2. Because the acceptance rule of §4 keeps a replacement only when it passes more of the same checks that compliance measures, these figures are the yield of the intervention rather than an independently identified effect. We have no systematic paired human evaluation of the changed outputs, so these gains do not demonstrate improved answer quality.

Figure 2 shows where the yield arises in the routed system: little on `qa-context`, which is already near ceiling, and more on the two generation subtasks, where first-pass output is weakest.

6.3 Ablations

LoRA. Run C uses 1.50% of the trainable parameters with no resolved aggregate advantage over full fine-tuning in the available paired comparisons. Its `sum-sum` outputs are also short (24 words held-

system	altered	first	shipped	yield
Run B alone	9.9%	0.9436	0.9715	+0.0279
Run A alone	9.2%	0.9374	0.9626	+0.0252
base alone	8.6%	0.9512	0.9749	+0.0237
routed	—	0.9554	0.9773	+0.0219

Table 4: Verification layer over all 12,775 test items. "altered" is the share of outputs that did not ship as first generated. "first" is after deterministic formatting cleanup but before retries and repairs; scores average per-item compliance. The routed row combines Run B's `qa-context` stream with base's other two subtasks.

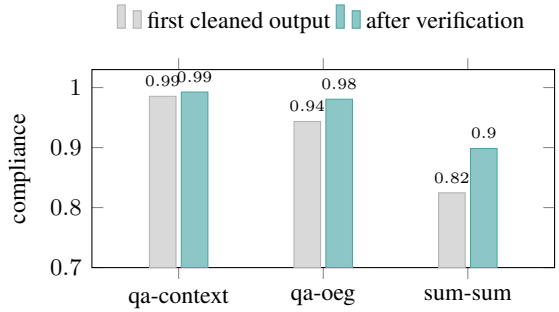

Figure 2: Compliance before and after verification for the routed system, following each subtask to the model that serves it: +0.007 on `qa-context` (from Run B), +0.037 on `qa-oeg` and +0.074 on `sum-sum` (both from base).

out against 25 for Runs A and B, versus 185 for base). This is consistent with adaptation to training targets whose median is about 25 words, but does not rule out other effects of either tuning recipe.

Language pin. A 288-item probe over both models shows the English pin helps despite being written in a different language from the prompt: compliance is 89.2% with it versus 86.9% without for Run B, and 90.0% versus 85.0% for base, concentrated in `sum-sum` where base compliance falls to 45.8% without it. This is a small-scale replication of the language-confusion effect documented by Marchisio et al. (2024), who find that models frequently generate in the wrong language and that explicit instruction mitigates it.

Augmentation. The sample-only Run A and augmented Run B show no resolved aggregate gain in the available paired comparisons, despite Run B's additional teacher-distilled, translated and native-BELEBELE data. This single-seed result does not establish that augmentation is generally ineffective. Yoruba's regression above is a separate, language-specific observation whose task mismatch prevents a direct augmentation claim.

7 Limitations

Compliance measures whether outputs are well-formed, not whether they are correct. Verification yield is measured using the same checks that select repairs, without paired human validation. The weak `qa-oeg` routing evidence and untested shared-base configuration further limit our deployment conclusions. All runs used one training seed: bootstrap intervals describe item-sampling uncertainty conditional on those runs, not variation across retraining. Both full fine-tunes peaked at epoch 2 and declined at epoch 3, showing checkpoint sensitivity rather than establishing seed instability. The recipe ordering and Yoruba pattern have unknown sensitivity to training randomness.

The base held-out score 0.8073 and first-pass test score 0.9512 differ by 14.39 points, but are not the same estimand. Held-out scoring gives the length pass rate (47/169, or 0.2781) one quarter of the pooled score, even though only 4.7% of items have a scored constraint; test scoring averages per-item checks (Appendix A). Task proportions also shift: `qa-context` rises from 41.0% to 67.6% and `sum-sum` falls from 37.1% to 13.9%, while parsed constraint coverage rises to 85.6%. Thus more constraints need not lower the reported score. Aggregation and distribution changes preclude attributing the gap to model improvement; the retained aggregates cannot isolate their individual contributions or predict performance on a new test set.

8 Conclusion

We submitted a routed system to WMT26 MIST: fine-tuned and base behaviour under a fixed task lookup with four verification checks. Our submitted outputs used 9.46B parameters; the proposed 5.16B shared-base LoRA configuration remains untested end-to-end.

Fine-tuning improves context-based QA substantially through output-format acquisition. Verification alters roughly 9% of outputs and adds 2.2 compliance points as a yield of its acceptance rule, without demonstrated answer-quality gains. The available single-seed comparisons show no resolved aggregate benefit from augmentation or full fine-tuning over LoRA. Yoruba's regression and partial recovery highlight the need for task-specific data audits and uncertainty estimates before generalizing these observations.

Acknowledgements

We thank the WMT26 MIST organizers for the shared task data and for the prompt fixes issued during the test release period.

References

Lucas Bandarkar, Davis Liang, Benjamin Muller, Mikel Artetxe, Satya Narayan Shukla, Donald Husa, Naman Goyal, Abhinandan Krishnan, Luke Zettlemoyer, and Madian Khabsa. 2024. The BELEBELE benchmark: A parallel reading comprehension dataset in 122 language variants. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 749–775.

Abhik Bhattacharjee, Tahmid Hasan, Wasi Uddin Ahmad, Yuan-Fang Li, Yong-Bin Kang, and Rifat Shahriyar. 2023. CrossSum: Beyond English-centric cross-lingual summarization for 1,500+ language pairs. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 2541–2564.

Laurie Burchell, Alexandra Birch, Nikolay Bogoychev, and Kenneth Heafield. 2023. An open dataset and model for language identification. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 865–879.

Pinzhen Chen and WMT26 MIST Organizers. 2026. WMT26 MIST shared task sample training data (22 june 2026 revision). `https://huggingface.co/datasets/pinzhenchen/wmt26-mist-sample`. Dataset.

Jonathan H. Clark, Eunsol Choi, Michael Collins, Dan Garrette, Tom Kwiatkowski, Vitaly Nikolaev, and Jennimaria Palomaki. 2020. TyDi QA: A benchmark for information-seeking question answering in typologically diverse languages. *Transactions of the Association for Computational Linguistics*, 8:454–470.

Patrick Fernandes, António Farinhas, Ricardo Rei, José G. C. de Souza, Perez Ogayo, Graham Neubig, and André F. T. Martins. 2022. Quality-aware decoding for neural machine translation. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*, pages 1396–1412.

Thomas Palmeira Ferraz, Kartik Mehta, Yu-Hsiang Lin, Haw-Shiuan Chang, Shereen Oraby, Sijia Liu, Vivek Subramanian, Tagyoung Chung, Mohit Bansal, and Nanyun Peng. 2024. LLM self-correction with DeCRIM: Decompose, critique, and refine for enhanced following of instructions with multiple constraints. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Markus Freitag, David Grangier, Qijun Tan, and Bowen Liang. 2022. High quality rather than high model probability: Minimum Bayes risk decoding with neural metrics. *Transactions of the Association for Computational Linguistics*, 10:811–825.

Gemma Team, Google DeepMind. 2025. Gemma 3 technical report. *Preprint*, arXiv:2503.19786.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations (ICLR)*.

Jie Huang, Xinyun Chen, Swaroop Mishra, Huaixiu Steven Zheng, Adams Wei Yu, Xinying Song, and Denny Zhou. 2024. Large language models cannot self-correct reasoning yet. In *International Conference on Learning Representations (ICLR)*.

Armand Joulin, Edouard Grave, Piotr Bojanowski, Matthijs Douze, Hérve Jégou, and Tomas Mikolov. 2016. FastText.zip: Compressing text classification models. *Preprint*, arXiv:1612.03651.

Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. 2017. Bag of tricks for efficient text classification. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics (EACL)*, pages 427–431.

Amir Hossein Kargaran, Ayyoob Imani, François Yvon, and Hinrich Schütze. 2023. GlotLID: Language identification for low-resource languages. In *Findings of the Association for Computational Linguistics: EMNLP 2023*.

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 Schmidtova, 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 (WMT)*, Suzhou, China.

Philipp Koehn. 2004. Statistical significance tests for machine translation evaluation. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 388–395.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. 2023. Efficient memory management for large language model serving with PagedAttention. In *Proceedings of the 29th Symposium on Operating Systems Principles (SOSP)*, pages 611–626.

Faisal Ladhak, Esin Durmus, Claire Cardie, and Kathleen McKeown. 2020. WikiLingua: A new benchmark dataset for cross-lingual abstractive summarization. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 4034–4048.

Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, Shashank Gupta, Bodhisattwa Prasad Majumder, Katherine Hermann, Sean Welleck, Amir Yazdanbakhsh, and Peter Clark. 2023. Self-refine: Iterative refinement with self-feedback. In *Advances in Neural Information Processing Systems (NeurIPS)*.

Kelly Marchisio, Wei-Yin Ko, Alexandre Bérard, Théo Dehaze, and Sebastian Ruder. 2024. Understanding and mitigating language confusion in LLMs. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6653–6677, Miami, Florida, USA.

NLLB Team, Marta R. Costa-jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, et al. 2022. No language left behind: Scaling human-centered machine translation. *Preprint*, arXiv:2207.04672.

Sara Papi, Marco Gaido, Andrea Pilzer, Matteo Negri, et al. 2025. MCIF: Multimodal crosslingual instruction-following benchmark from scientific talks.

Maja Popović. 2017. chrF++: Words helping character n-grams. In *Proceedings of the Second Conference on Machine Translation (WMT)*, pages 612–618.

Matt Post. 2018. A call for clarity in reporting BLEU scores. In *Proceedings of the Third Conference on Machine Translation (WMT)*, pages 186–191.

Ricardo Rei, Craig Stewart, Ana C Farinha, and Alon Lavie. 2020. COMET: A neural framework for MT evaluation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2685–2702.

Shivalika Singh, Freddie Vargus, Daniel D'souza, Börje F. Karlsson, Abinaya Mahendiran, Wei-Yin Ko, Herumb Shandilya, Jay Patel, Deividas Mataciunas, Laura OMahony, Mike Zhang, Ramith Hettiarachchi, Joseph Wilson, Marina Machado, Luisa Souza Moura, et al. 2024. Aya dataset: An open-access collection for multilingual instruction tuning. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 11521–11567.

A Metric definitions

Checks and aggregation. For a nonempty output on item i, let L_i, D_i and A_i indicate language non-failure, non-degeneracy and absence of chat-template artefacts. Unscorable language identification counts as non-failure, not as a confirmed language match or an omitted check. Let I_i indicate a parsed, scorable length constraint and H_i its binary outcome. The test analysis and held-out bootstrap use

$$s_i = \frac{L_i + D_i + A_i + I_i H_i}{3 + I_i}, \qquad C_{\text{item}} = \frac{1}{n} \sum_i s_i,$$

with empty outputs assigned zero. An unavailable length verdict omits that check. Constraint coverage therefore affects the effective weights.

The historical held-out ladder instead pools counters before averaging check rates. With at least one scored length constraint,

$$C_{\text{pool}} = \tfrac{1}{4} \left(\bar{L} + \bar{D} + \bar{A} + \frac{\sum_i I_i H_i}{\sum_i I_i} \right).$$

Without one, it averages the first three rates. Its implementation counts empty outputs in n but skips their checks, so they do not register as failures in the first three rates, unlike the per-item score. The native selected-checkpoint slices reported here contain no empty outputs. Table 3 pools over the entire native slice; Table 5 instead averages language–task cell pooled scores weighted by cell size. Its "all" entry is the global ladder score, not a weighted mean of the displayed language scores. These aggregations are not interchangeable: base's global score is $(0.9932 + 0.9580 + 1 + 0.2781)/4 \simeq 0.8073$. We preserve the historical values and identify their definitions rather than applying the per-item bootstrap intervals to pooled scores. The selection aggregate remains $\text{agg} = C_{\text{pool}} + \text{F}_1^{\texttt{qa-context}}$, the pre-specified selection rule, not an independent quality measure.

Parser and repetition detector. The length parser matches localized number and unit forms, including Unicode digits and Yoruba postposed numbers and a spelled-out hundred. It scopes matching to an inferred instruction region; document numbers can still be mistaken for constraints. Ranges use their midpoint, approximate counts allow $\pm 30\%$, and maximum constraints use an upper bound. Missing or unscorable constraints are omitted. Sentence counting counts terminal punctuation characters, not syntactically parsed sentences; abbreviations and repeated punctuation can distort it. Repetition detection flags an exact whitespace-delimited 4-gram occurring at least four times; unspaced Chinese, Japanese and Thai text uses character 6-grams with the same threshold. Legitimate recurring phrases can trigger this heuristic, and it does not assess Yoruba tone-mark or orthographic

correctness. These checks are proxies, not a substitute for human evaluation in low-resource languages.

B Per-language compliance

Table 5 gives compliance per language on the native held-out slice, aggregated over subtasks and weighted by item count. We report held-out rather than test figures because the per-item test artefacts were not retained; the held-out set covers 27 languages, all but Bhojpuri of the 24 test languages. Per-language confidence intervals were not computed, and the per-item generations required to reconstruct them were not retained. The aggregates do not support recovering paired intervals; these language-level comparisons are descriptive.

lang	n	base	B	C	lang	n	base	B	C
arb	214	0.916	0.909	0.845	mar	138	0.911	0.905	0.904
ben	77	0.747	0.750	0.750	pes	101	0.861	0.861	0.861
ces	7	1.000	1.000	0.964	por	161	0.881	0.901	0.906
ckb	45	1.000	1.000	1.000	rus	213	0.927	0.942	0.937
deu	152	0.850	0.977	0.967	slk	45	1.000	1.000	0.993
eng	157	0.970	0.959	0.941	spa	160	0.887	0.873	0.872
fin	45	0.744	0.750	0.750	swh	99	0.861	0.864	0.859
fra	158	0.888	0.876	0.878	tel	99	0.859	0.861	0.861
hat	45	0.985	0.993	0.993	tha	94	0.755	0.863	0.863
hin	168	0.808	0.838	0.857	tur	160	0.916	0.929	0.896
ind	213	0.878	0.873	0.872	vie	160	0.758	0.823	0.728
ita	153	0.994	0.975	0.957	yor	78	0.816	0.675	0.754
jpn	210	0.768	0.802	0.803	zho	199	0.802	0.857	0.865
kor	200	0.749	0.755	0.752	*all*	3551	0.807	0.824	0.815

Table 5: Compliance by language, native held-out slice, selected checkpoints. Language scores weight pooled task-cell scores by item count; “all” uses global pooling. Run A is omitted for compactness; its Yoruba score is 0.761. No per-language confidence intervals are available.