

HeyBusan at WMT26: A System for the Sorbian Track

Mikhail Khasykov

Independent

mkhasykov@gmail.com

Abstract

We describe our submission (team HeyBusan) to the Sorbian track of the WMT26 Multitask LLMs with Limited Resources shared task. A single Qwen3.5-2B model handles machine translation, question answering, spell checking, grammar checking, and mathematical reasoning in Upper Sorbian, Lower Sorbian, and German. We fine-tuned all model parameters on public data (the official task data plus external corpora) for approximately 55 GPU-hours on one 24 GB consumer GPU. Task-specific decoding then improved the results further without changing the model’s weights. Team HeyBusan placed first in machine translation, spell checking, and grammar checking, and tied with team LT3 for the overall track win.

1 Introduction

Can a single 2B model, trained on one consumer GPU, handle five tasks across both Sorbian languages? The WMT26 Multitask LLMs with Limited Resources shared task (Okabe et al., 2026) tested exactly this setting. Each team had to use one model for all five tasks, restricted to Qwen3.5-2B (Qwen Team, 2026) or a permitted smaller variant. The tasks cover Upper Sorbian (hsb), Lower Sorbian (dsb) and German: machine translation (MT), multiple-choice question answering (QA), spell checking (SC), grammar checking (GC), and mathematical reasoning (MR). Teams submitted the model’s *outputs* and could use different decoding methods for each task; all training data had to be publicly available. The final ranking gives equal weight to all five tasks through per-task rank points (Section 2).

Our submission was built under genuinely limited resources: a single RTX 3090 for everything, with the final model’s whole training path taking ≈ 55 GPU-hours. Our system finished joint first

with team LT3, placing first in three of the five tasks. Our contributions:

- a resource-constrained training recipe that handles all five tasks: full-parameter fine-tuning of one 2B model on one 24 GB GPU with public data;
- a task-specific decoding suite on top of the frozen model, led by QA option-order averaging.

2 Task setup and constraints

Tasks and ranking. Table 1 lists the five tasks, their official metrics and test sizes. The organizers scored the outputs submitted by each team; participants ran inference themselves with full control over prompting and post-processing. The final ranking uses per-task rank points: for each task, the best primary submission earns the maximum number of points. Scores within one point of each other are treated as tied, and teams with equal totals share the final rank.

Evaluation protocol. For model selection, we used three sources of development data: the untouched MT development set, held-out 20% subsets of QA/SC/GC, and the organizer-released 48-item MR set. The remaining 80% of QA/SC/GC entered training. The MR set is the Sorbian part of PluraMath (Dementieva et al., 2026), an extension of PolyMath (Wang et al., 2025) to 18 further languages; we used it only for evaluation and used no other PluraMath or PolyMath data. Over these *local validation* sets we averaged the five task metrics into the composite

$$S = \frac{1}{5} \left(\frac{\text{chrF}++}{100} + \text{QA} + \text{SC} + \text{GC} + \text{MR} \right).$$

Results from individual leaderboard submissions are labeled *leaderboard*. The leaderboard reports only per-task scores, so we computed S over those scores ourselves. The final results table uses the organizers’ *official* scores.

Task	Required output	Official metric	Test size
MT	translation inside <hsb>/<dsb> tags	chrF++ (Popović, 2017) over six directions	26,174
QA	option number (2–16 options)	accuracy	415
SC, GC	the incorrect word and its correction, or CORRECT in both fields	mean exact-match accuracy for detection and correction	SC 4,000 GC 3,726
MR	final answer inside <answer> tags	exact-string accuracy (no symbolic equivalence)	500

Table 1: The five tasks. The scorer extracts answers from the required tags; an output without these tags receives a score of zero. Scores are averaged over hsb and dsb (MT: the six directions among de/hsb/dsb).

3 Model training

Starting from Qwen3.5-2B, we ran 2,000 steps of multitask SFT, followed by a final 200-step pass that added the previously held-out QA/SC/GC and MT development examples to training. We then set the final weights to a weighted average of the two models — $0.9 \times$ this checkpoint + $0.1 \times$ the original Qwen3.5-2B — and applied per-task decoding (Section 4).

3.1 Training data (all public)

Table 2 lists the six components of the training mix; all source data are public. The external WMT20–22 parallel corpora (Fraser, 2020; Libovický and Fraser, 2021; Weller-Di Marco and Fraser, 2022) were decontaminated against all Sorbian-track MT test inputs and MT dev sentences before use, checking both sides of every candidate pair.¹ We used no data from the WMT25 edition of the task (see Section 6).

Core supervision came from the official task data: the full MT training set plus 80% of the QA/SC/GC development sets, with a hash-based 20% holdout reserved for local validation until we added it back during the final 200 steps. We sampled the parallel corpora into $\sim 435k$ directional MT examples, with at most 120k examples from each file.

SC/GC synthesis. The official development sets provide only 1,250–2,000 examples per task

¹For example, the WMT22 de-hsb training split contained 7,334 exact matches to MT test inputs.

Source	How it was used
Official task data (MT train; 80% of QA/SC/GC dev, 20% held out; 100% of QA/SC/GC dev plus MT dev in the final 200 steps)	core supervision
WMT20–22 de-hsb / de-dsb / hsb-dsb (Fraser, 2020; Libovický and Fraser, 2021; Weller-Di Marco and Fraser, 2022) (+304k/+37k/+61k pairs, decontaminated)	main MT training pairs
FineWeb-2 hsb/dsb (Penedo et al., 2025) (12.9M/1.9M words)	additional monolingual text
Instruction mix, $\sim 32k$ rows (Singh et al., 2024; Longpre et al., 2023; Zhao et al., 2024; Lambert et al., 2024; Köpf et al., 2023; Open Assistant contributors, 2023)	general instruction data to reduce forgetting, following (Purason and Fishel, 2025)
Dataset generated from the soblex / dolnoserski full-form lexicons ($\sim 20k$ rows/task/lang)	synthetic spelling and grammar examples
Reasoning chains for German GSM8K-train questions (Cobbe et al., 2021; Zelikman et al., 2022) (2,657 model chains + 785 gold solutions)	filtered examples of mathematical reasoning

Table 2: Training data.

and language, so we generated additional examples from correct sentences in the official monolingual corpora. Both tasks build on the full-form lexicons: dictionaries that list every recorded form of a word, such as all its case and number forms.

For GC, we replaced one word with a different form of the same word (for example, with a different case, number, or gender ending). Both the original word and its replacement had to belong to exactly one entry in the lexicon. In 70% of cases we took a word right after a preposition or next to a word it must agree with, because there the ending is fixed by the context.

For SC, we changed one word by one or two character edits (one edit in 80% of cases): a diacritic change, or a replaced, swapped, deleted, inserted, or doubled letter. We kept an edit only if the result is not a word: it must be absent from the lexicon and rejected by the hunspell spell checker.

For both tasks, we left about half of the sentences unchanged and labeled them CORRECT, teaching the model not to invent an error in a correct sentence.

MR synthesis. We translated GSM8K training questions (Cobbe et al., 2021) into German with the original Qwen3.5-2B, because the model han-

dles German far better than Sorbian and because all our parallel data pair German with Sorbian. An earlier checkpoint of our own model (from a QLoRA run, before we switched to full fine-tuning) then sampled eight reasoning chains per question at temperature 0.7. Following STaR (Zelikman et al., 2022), we kept at most one chain per question, and only if its final answer matched the gold answer; this retained 2,657 chains. For the 785 questions where no sampled chain reached the gold answer, we used the original English GSM8K solution as the target instead, so for these questions the target is in English while the question stays in German.

3.2 Training regime

We used full-parameter bf16 SFT with packed 1,536-token sequences, an effective batch size of 32, paged 8-bit AdamW, and a cosine learning-rate schedule with a peak of 7×10^{-5} . Training comprised 2,000 main steps and the final 200-step pass, taking ≈ 55 GPU-hours in total on one 24 GB RTX 3090.

How we evaluated training choices. Because 80% of the QA/SC/GC development examples entered training, their full development scores were contaminated; we therefore selected models on held-out local-validation subsets. The merge and the final 200-step pass were each checked with one leaderboard submission, which is not enough to claim a real improvement (Section 6). In total we used 13 of the 20 allowed submission rounds. Table 3 summarizes the five training decisions with one primary number each.

Full fine-tuning clearly beat our tuned QLoRA baseline (all-linear rank-64, NF4-quantized base) with the data mixture, step count and batch size held fixed, and the gain was confirmed on held-out validation data — consistent with what team NRC reported at WMT25 (Larkin et al., 2025). The base-weight merge ($0.1 \cdot \text{base} + 0.9 \cdot \text{fine-tune}$) recovered some QA and math ability (leaderboard QA +2.7 pp, MR +3.2 pp) while slightly reducing the other tasks (MT −0.2 chrF++, SC −1.2 pp, GC −0.1 pp), echoing team TartuNLP’s WMT25 merging observation (Purason and Fishel, 2025). The released final weights are at <https://huggingface.co/mkhasikov/qwen3.5-2b-wmt26-sorbian>.

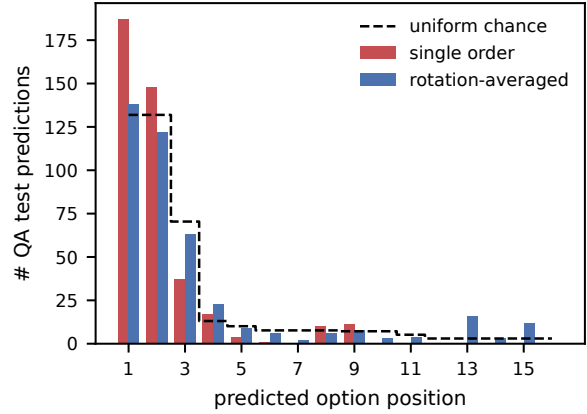


Figure 1: Predicted answer position on the QA test set. With the original option order (red), predictions collapse onto early positions; rotation averaging (blue) moves them toward the dashed line — the distribution expected if every available answer were equally likely.

4 Per-task decoding

The metric structure guided decoding (Table 4): MT’s graded chrF++ score favored beam search, whereas SC, GC, and MR benefited from sampling votes; SC and GC also benefited from lexical constraints.

QA position debias. The model favored answers near the beginning of the option list. With the original option order, it never selected an answer beyond position 9, even though 92 of the 415 test questions had at least 10 options (Figure 1). We therefore rotated each option list so that every answer appeared once in every position. For each answer, we averaged its log-softmax score across these rotations and selected the highest average. This increased accuracy by 6.9 percentage points on local validation ($n=58$) and by **6.5 percentage points on the leaderboard**. Team JGU Mainz reported a similar benefit from permutation-based likelihood averaging at WMT25 (Saadi et al., 2025).

Lexicon vote filters. For SC and GC, we generated 16 corrections per sentence and selected the majority prediction (Wang et al., 2023). Before voting, we removed any sample whose proposed correction did not appear in the lexicon. If this removed all 16 samples, we retained the unfiltered set. We applied this filter to both languages for SC and only to Lower Sorbian for GC; the incomplete Upper Sorbian lexicon reduced local-validation accuracy.

MR in two languages. For each MR question we first let our model translate the question into

Decision	Primary evidence	Caveat
Full fine-tuning instead of QLoRA (Dettmers et al., 2023)	local S +0.111	the comparison also changes quantization and learning rate
External-data bundle (parallel + monolingual + instructions + reasoning pool)	local MT +8.54 chrF++	the four components were not ablated separately
Lexicon-based SC/GC synthesis	local SC +3.0 pp; GC +2.1 pp	the data recipe and task balance changed together
Weight averaging with the base model	leaderboard S +0.009	one leaderboard submission; $w=0.1$ was chosen by a local-validation sweep
Final 200-step pass (adds all QA/SC/GC and MT development examples; never MR dev)	leaderboard S +0.003	one leaderboard submission; cannot separate extra steps from the added data

Table 3: The five training decisions and the primary evidence behind each. The decisions were made in this order, each on top of the previous ones; every number compares the model with that decision against the model without it, all other settings unchanged. “Local” numbers are on local validation; “leaderboard” numbers come from one leaderboard submission each.

Task	Final decoding	Evidence
MT	beam search, 4 beams	local MT +1.5 chrF++
QA	argmax after averaging option log-likelihoods over all rotations of the option order	leaderboard QA +6.5 pp
SC	majority vote over 16 answers sampled at $T=0.7$ (each answer: the detected word and its correction), after removing answers whose correction is not in the lexicon	leaderboard SC +3.0 pp
GC	the same vote, with the lexicon filter applied to Lower Sorbian sentences only	leaderboard GC +0.4 pp
MR	majority vote over 32 samples: 16 from the Sorbian question and 16 from its German translation by our model	local MR +14.6 pp

Table 4: Final per-task decoding. Each delta compares the final decoding with greedy decoding of the same model.

German. We then sampled 16 answers at $T=0.7$ from the Sorbian question and 16 from the German translation, and took the majority answer over all 32 samples. If the translation lost or changed a number from the question, we discarded it and sampled all 32 answers from the Sorbian question instead.

Inference cost. All test outputs were generated on the same RTX 3090 with Hugging Face transformers in bf16. With greedy decoding, all five test sets take about 2–3 GPU-hours. The final decoding is several times more expensive: beam search with 4 beams over the 26,174 MT inputs, 16 samples for each of the 7,726 SC/GC sentences, 32 samples for each of the 500 MR questions (16 for the Sorbian question and 16 for its German translation), and up to 16 option rotations per QA question. In total this took roughly 10–15 GPU-hours, dominated by MR sampling and MT beam search.

5 Results

Table 5 shows the official final results. HeyBusan led MT, SC and GC, while team LT3 led QA and MR; the two teams tied at 22 rank points and were declared **joint winners of the Sorbian track**. Official SC/GC scores are computed differently from the leaderboard’s display — the official metric averages detection and correction exact match — so the official and leaderboard numbers are not directly comparable.

6 Limitations

A single 24 GB GPU limited the study: each training comparison ran once, with no repeated seeds. The QA/SC/GC local holdouts were small (QA $n=58$) and did not always predict leaderboard behavior. The merge and the final 200-step pass were each evaluated with one leaderboard submission, so their small gains may simply be noise. We used

Team	MT	QA	SC	GC	MR	Pts
HeyBusan	72.75	55.92	78.03	79.52	29.00	22
LT3	69.57	65.05	76.11	78.32	30.40	22
Hitsz	63.12	57.33	73.13	62.11	18.60	16
HSE	61.50	46.77	63.63	50.02	5.00	9
baseline	21.68	43.39	6.66	1.34	5.40	5

Table 5: Official final results of the Sorbian track (Okabe et al., 2026). Primary team submissions are shown with the organizers’ baseline for reference; the best team score per task is in bold. Teams HeyBusan and LT3 finished level at 22 rank points and are the joint winners of the track.

13 of the 20 submission rounds and did not tune continuous parameters on the test set.

MR remained our weakest task (official accuracy 29.00) despite the synthetic reasoning data and the two-view decoding. We also overlooked the parallel and monolingual corpora released for the WMT25 edition of the task (about 190k de-hsb and 170k de-dsb sentence pairs), although the rules allowed them; this was an oversight on our part.

7 Conclusion

With one 2B model, one consumer GPU, and public data, HeyBusan was a joint winner of the WMT26 Sorbian track. The successful recipe combined full-parameter fine-tuning, additional public data, and task-specific decoding, led by QA option-order averaging.

Acknowledgments

We thank the organizers of the shared task. The soblex (<https://www.soblex.de>; GPL-3.0) and dolnoserbski.de (<https://dolnoserbski.de>) full-form lexicons (both accessed July 2026) were used as inference-time filters and synthetic-data generators and are not redistributed. Other resources: WMT20–22 corpora (statmt.org), FineWeb-2, GSM8K-train (MIT), Qwen3.5-2B (Apache-2.0).

References

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*.

Daryna Dementieva, Nikolay Babakov, Kathy Hämmerl, Ilseyar Alimova, Jindřich Libovický, Shu Okabe, et al. 2026. *PluraMath: Extending mathematical reasoning evaluation beyond high-resource languages*. *arXiv preprint arXiv:2607.05992*.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. 2023. QLoRA: Efficient finetuning of quantized LLMs. In *Advances in Neural Information Processing Systems*.

Alexander Fraser. 2020. *Findings of the WMT 2020 shared tasks in unsupervised MT and very low resource supervised MT*. In *Proceedings of the Fifth Conference on Machine Translation*, pages 765–771, Online.

Andreas Köpf, Yannic Kilcher, Dimitri von Rütte, et al. 2023. OpenAssistant conversations — democratizing large language model alignment. In *Advances in Neural Information Processing Systems*.

Nathan Lambert et al. 2024. Tulu 3: Pushing frontiers in open language model post-training. *arXiv preprint arXiv:2411.15124*.

Samuel Larkin, Chi-Kiu Lo, and Rebecca Knowles. 2025. *NRC systems for the WMT2025-LRSL shared task*. In *Proceedings of the Tenth Conference on Machine Translation*, pages 1133–1142, Suzhou, China.

Jindřich Libovický and Alexander Fraser. 2021. *Findings of the WMT 2021 shared tasks in unsupervised MT and very low resource supervised MT*. In *Proceedings of the Sixth Conference on Machine Translation*, pages 726–732, Online.

Shayne Longpre et al. 2023. The flan collection: Designing data and methods for effective instruction tuning. In *International Conference on Machine Learning*.

Shu Okabe, Marion Di Marco, Kathy Hämmerl, Daryna Dementieva, Lukas Edman, Marko Měškank, Anita Hendrichowa, and Alexander Fraser. 2026. Findings of the WMT26 Shared Task: Multitask LLMs with Limited Resources. In *Proceedings of the Eleventh Conference on Machine Translation*, Budapest, Hungary. Association for Computational Linguistics.

Open Assistant contributors. 2023. OASST2 dataset. <https://huggingface.co/datasets/OpenAssistant/oasst2>.

Guilherme Penedo, Hynek Kydlíček, Vinko Sabolčec, Bettina Messmer, Negar Foroutan, Amir Hossein Kargaran, Colin Raffel, Martin Jaggi, Leandro von Werra, and Thomas Wolf. 2025. FineWeb2: One pipeline to scale them all — adapting pre-training data processing to every language. *arXiv preprint arXiv:2506.20920*.

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

- Taido Purason and Mark Fishel. 2025. [TartuNLP at WMT25 LLMs with limited resources for Slavic languages shared task](#). In *Proceedings of the Tenth Conference on Machine Translation*, pages 1143–1150, Suzhou, China.
- Qwen Team. 2026. [Qwen3.5-2B model card](#).
- Hossain Shaikh Saadi, Minh Duc Bui, Mario Sanz-Guerrero, and Katharina von der Wense. 2025. [JGU Mainz’s submission to the WMT25 shared task on LLMs with limited resources for Slavic languages: MT and QA](#). In *Proceedings of the Tenth Conference on Machine Translation*, pages 1151–1157, Suzhou, China.
- Shivalika Singh, Freddie Vargus, Daniel D’souza, Börje F. Karlsson, 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 (Volume 1: Long Papers)*, pages 11521–11567.
- Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2023. Self-consistency improves chain of thought reasoning in language models. In *International Conference on Learning Representations*.
- Yiming Wang, Pei Zhang, Jialong Tang, Haoran Wei, Baosong Yang, et al. 2025. [PolyMath: Evaluating mathematical reasoning in multilingual contexts](#). *arXiv preprint arXiv:2504.18428*.
- Marion Weller-Di Marco and Alexander Fraser. 2022. [Findings of the WMT 2022 shared tasks in unsupervised MT and very low resource supervised MT](#). In *Proceedings of the Seventh Conference on Machine Translation*, pages 801–805.
- Eric Zelikman, Yuhuai Wu, Jesse Mu, and Noah D. Goodman. 2022. STaR: Bootstrapping reasoning with reasoning. In *Advances in Neural Information Processing Systems*.
- Wenting Zhao, Xiang Ren, Jack Hessel, Claire Cardie, Yejin Choi, and Yuntian Deng. 2024. WildChat: 1m ChatGPT interaction logs in the wild. In *International Conference on Learning Representations*.