

Zolint at WMT 2026: A Multitask LLM Submission for the Low-Resource Ukrainian Track

Lingchu Yang^{1*} Zitong Fu^{2*} Kathy Hämmerl^{2,3} Masaki Ito¹

¹Graduate School of Information Science and Technology, The University of Tokyo

²School of Computation, Information and Technology, Technical University of Munich

³Munich Center for Machine Learning

Abstract

We present a unified multi-task fine-tuning system submitted to the WMT 2026 Shared Task on Multitask LLMs with Limited Resources (Ukrainian track), covering machine translation, spelling correction, grammar correction, question answering, and mathematical reasoning. Our system is built on Qwen3.5-2B-Instruct and adopts a two-stage training pipeline that gradually introduces heterogeneous tasks while replaying selected data from earlier stages to mitigate catastrophic forgetting. The resulting model achieves a strong balance across all evaluated tasks, consistently outperforming the official baseline and obtaining the best performance on every task except question answering. Our results demonstrate that staged multi-task fine-tuning is an effective strategy for adapting compact multilingual LLMs under limited-resource constraints.

1 Introduction

Large language models (LLMs) have achieved remarkable success on high-resource languages, yet their performance remains considerably weaker for low-resource languages such as Ukrainian. The WMT 2026 Shared Task: Multitask LLMs with Limited Resources investigates whether a single Qwen3.5 model with at most 2B parameters can jointly solve diverse Ukrainian NLP tasks, including machine translation, spelling correction, grammar correction, question answering, and mathematical reasoning.

This setting poses a challenging multi-task learning problem. These tasks require different capabilities, ranging from cross-lingual generation and linguistic correction to language understanding and multi-step reasoning. Jointly optimizing such heterogeneous objectives often leads to task interference and catastrophic forgetting, especially for compact language models with limited capacity.

To address these challenges, we propose a two-stage multi-task fine-tuning pipeline. The first stage focuses on English–Ukrainian translation together with spelling and grammar correction to establish fundamental Ukrainian language abilities. The second stage introduces Czech–Ukrainian translation, question answering, and mathematical reasoning while replaying a subset of first-stage data to mitigate catastrophic forgetting.

Besides the staged training strategy, we construct additional spelling correction data from the question answering corpus and observe a consistent positive interaction between the two tasks, suggesting that auxiliary correction supervision benefits language understanding in low-resource settings. We also find grammar correction to be most susceptible to forgetting after continual multi-task training.

Our unified multi-task system achieves a strong balance across all evaluated tasks. Compared with the official baseline, it consistently improves overall performance and achieves the best results on all tasks except question answering. These results demonstrate that carefully scheduling heterogeneous tasks and mitigating forgetting are crucial for adapting compact multilingual LLMs in limited-resource settings. We release our model and code.¹

2 Related Work

2.1 Low-resource Multilingual LLMs

Despite modern LLMs’ strong multilingual capabilities, their performance on low-resource languages still has substantial room for improvement due to the imbalance of pre-training data. Recent work has explored continued instruction tuning, synthetic data generation, and bilingual supervision to improve multilingual performance, yet significant performance gaps remain for low-resource languages

¹Github: <https://github.com/vanessafu/nlp-2026-low-resource-ukrainian-mt-qa>;
Model: <https://huggingface.co/MDK-YLC/Qwen3.5-2B-WMT26-Ukrainian-LoRA>

*Equal contribution.

Task	Train Set	Dev Set	Description
MT (EN→UK)	95,140 pairs	5,108 pairs	Only provided sentence-level dev set. Constructed both sentence- and paragraph-level training data.
MT (CS→UK)	16,946 pairs	9,424 pairs	Only provided sentence-level dev set. Constructed sentence- and paragraph-level training data and augmented dev set.
QA (ZNO)	7,379 pairs	613 questions	Augment the training set by shuffling, introducing CoT and comprehension understanding
QA (MMLU)	21,531 questions	285 questions	Augmented training set with back-translated MMLU data
GC	32,036 sentences	2,000 sentences	Only provided dev set. Constructed training set by UA-GEC
SC	6,232 sentences	2,000 sentences	Only provided dev set. Built training set by ourselves
MR	14,895 questions	24 questions	Only provided dev set. Constructed training set by back-translating English data

Table 1: Summary of the datasets for the five tasks.

such as Ukrainian (Okabe et al., 2025). The analysis of the WMT 2025 Shared Task systems (Okabe et al., 2025) also revealed difficulties in maintaining complete paragraph-level information during translation, suggesting that context preservation remains an important challenge for low-resource multilingual LLMs.

2.2 Multi-task Instruction Tuning

Instruction tuning enables a single LLM to solve diverse downstream tasks through natural language instructions (Chung et al., 2022). However, jointly optimizing heterogeneous tasks often introduces task interference and catastrophic forgetting, particularly for compact models (Luo et al., 2025). The WMT 2025 Shared Task further reports that optimizing translation alone degrades question answering performance, while unrestricted reasoning may negatively affect translation quality (Okabe et al., 2025). These findings motivate staged training strategies that balance linguistic knowledge and reasoning ability during multi-task adaptation.

3 Data

In this section, we describe the datasets used for the five tasks. For each task, we introduce the data source, explain the processing strategies, and report the resulting data sets. To enable joint instruction tuning, all datasets are converted into a unified instruction-response format. An overview of the dataset is shown in Table 1.

Machine Translation (MT). For EN→UK, we start from the WMT 2025 constrained parallel corpora (ParaCrawl, WikiMatrix, TED, ELRC, and OPUS). Following quality filtering, we apply SDKM-style similarity retrieval (Saadi et al., 2025)

using Qwen2.5-3B-Instruct embeddings, obtaining 108k high-quality sentence pairs. We further cluster semantically similar sentences with MiniBatchK-Means to construct 42k pseudo-document training examples for long-context translation.

For CS→UK, we combine public parallel corpora, including NLLB (Team et al., 2022) and OPUS resources (ELRC, Wikimedia, and EUBookshop) (Tiedemann, 2012). To construct paragraph-level data, we use the monolingual corpus from UberText Ukrainian fiction (Chaplynskyi, 2023). Then we segment the text into sentences using Stanza, group consecutive sentences into paragraphs of 300–700 words, and then back-translate them with the LINDAT Ukrainian→Czech system (Popel et al., 2024), yielding together with sentence-level data 16,946 training pairs in total.

All MT data are processed with a unified cleaning pipeline including quality filtering, normalization, deduplication, FastText language identification (Grave et al., 2018), multilingual-e5-small semantic filtering (Wang et al., 2024), and final quality estimation.

Question Answering (QA). The official QA dataset is constructed from two sources: the ZNO dataset (Romanyshyn et al., 2024) and the Ukrainian MMLU dataset (Yukhymenko et al., 2026) provided in the WMT2026 shared task. In addition to these official resources, we further enhance the QA data as follows. We first oversample the ZNO training set by a factor of two and randomly shuffle answer options to reduce position bias. We then incorporate additional CoT examples from the training split of NLPForUA/dumy-zno-ukrainian-math-history-geo-r1-o1 (Syromiatnikov and team, 2025). To ensure data integrity and avoid

contamination, we aligned this data with the official training set, filtered out any overlaps with the official development set, and retained only valid training instances with `o1_score = 1`. A deterministic 70%/30% split defines implicit and explicit CoT formats: in implicit CoT, the reasoning trace is included in the input and only the final answer is produced; in explicit CoT, the model generates full reasoning followed by the final answer. For MMLU, we further extend coverage using training split of the `cais/mmlu` dataset (Hendrycks et al., 2021a). We used Google Translate API to translate English questions into Ukrainian and merged them into the same QA format.

Grammar Checking (GC). GC training data are derived from **UA-GEC** (Syvokon and Romanyshyn, 2024), using the `gec-only` annotation layer. Each source–target sentence pair is converted into the shared-task format by aligning erroneous and corrected spans. We keep only sentences with at most one error region, yielding 32,036 training instances. The official GC development set (2,000 instances) is taken from the WMT 2026 Ukrainian shared-task release.

Spell Checking (SC). No dedicated Ukrainian spell-checking training corpus was provided by the organizers. We therefore construct SC training data synthetically from **MMLU_UKR** (Yukhymenko et al., 2026): Ukrainian question stems and answer options are taken from the released validation split, and spelling errors are injected automatically (e.g., character swaps, deletions, duplications, and visually confusable Cyrillic/Latin substitutions). About half of the sentences are left unchanged and labeled **CORRECT**. This yields 6,232 training instances. The SC development set (2,000 instances) is released by the organisers.

Math Reasoning (MR). MR training data come from two English math datasets separate from the official MR evaluation benchmark: **GSM8K** (Cobbe et al., 2021) (7,473 problems) and **Competition Math** (the MATH dataset) (Hendrycks et al., 2021b) (7,500 problems). We used Google Translate API to back-translate problem statements and reference solutions from English to Ukrainian. Then we cleaned the data to enhance data quality, including converting decimal numbers to equivalent $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ format fractions, correcting corrupted answer annotations, enforcing a single final boxed answer, and correct-

ing examples with invalid $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$. We also standardize all targets to the format `Answer: \boxed{...}`. Overall, 78 corrupted instances are removed, leaving 14,895 training examples. The official MR development set (24 problems: 12 low-difficulty and 12 medium-difficulty) is released by the WMT 2026 organizers.

4 System Architecture & Methodology

All tasks are handled by a single LoRA-finetuned Qwen3.5-2B-Instruct model (Hu et al., 2021) without any task-specific parameters, strictly satisfying the parameter constraint of the shared task. As illustrated in Figure 1, we propose a two-stage fine-tuning framework to adapt the foundation model across multiple tasks, which achieves optimal overall performance.

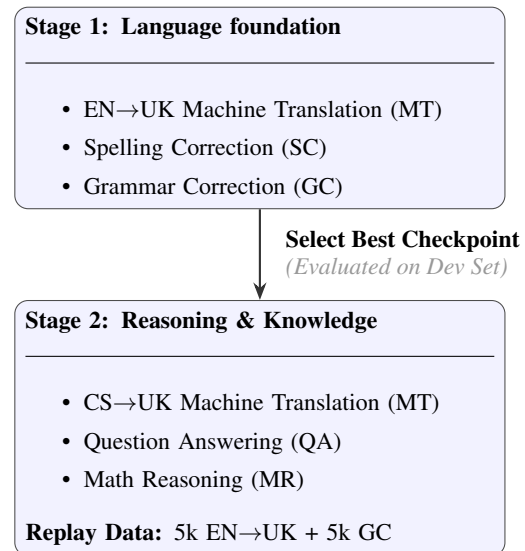


Figure 1: Overview of the simplified two-stage multi-task training framework.

4.1 Stage 1: EN→UK MT, SC, and GC

We first fine-tune the model on English→Ukrainian machine translation (MT), spelling correction (SC), and grammar correction (GC). The objective of this stage is to strengthen Ukrainian generation and correction ability before introducing more heterogeneous tasks.

For MT, the model is instructed to translate English sentences into Ukrainian and output only the translation. For SC and GC, we adopt a unified decision-then-correction format. The model first determines whether the input sentence is correct. If no error exists, it outputs **CORRECT**; otherwise, it predicts the incorrect word and its correction in

Model	Machine Translation (MT)				Spelling Correction (SC)		Grammar Correction (GC)		QA	MR
	EN→UK		CS→UK		EM-W	EM-C	EM-W	EM-C	Acc.	Acc.
	BLEU	chrF++	BLEU	chrF++						
Baseline	9.83	30.78	11.89	34.16	0.1745	0.1725	0.0425	0.0255	0.3485	0.1250
Joint	18.51	41.91	21.86	46.06	0.7520	0.6065	0.4425	0.4370	0.3719	0.1667
2-Stage-I	18.72	42.17	21.45	45.76	0.7610	0.6140	0.3820	0.3780	0.3820	0.2500
2-Stage-II	18.74	42.25	21.65	45.90	0.5390	0.4690	0.4135	0.4095	0.3686	0.2083

Table 2: Performance on development-set. EM-W and EM-C denote exact match on incorrect and corrected words, respectively.

the format {wrong word} -> {correct word}. During evaluation, predictions are converted into corresponding word pairs, while correct sentences are mapped to (CORRECT, CORRECT).

The first-stage model is trained for five epochs, with the checkpoint selected according to development performance. However, we observe that the best checkpoint is usually obtained after the first epoch, whereas later epochs often lead to overfitting under the low-resource setting.

4.2 Stage 2: CS→UK MT, QA, and MR

Starting from the best checkpoint from the first stage, we continue joint multi-task training by introducing Czech→Ukrainian machine translation, multiple-choice question answering (QA), and mathematical reasoning (MR).

Notably, we split the two MT directions and move the CS→UK task to the second stage for two motivations. First, as Czech and Ukrainian are closely related Slavic languages, CS→UK translation may benefit from a richer representational foundation established during Stage 1. Second, training CS→UK in Stage 2 places it closer to the final checkpoint, thereby mitigating catastrophic forgetting by reducing its exposure to non-MT fine-tuning in subsequent steps.

For CS→UK MT, the model translates Czech text into Ukrainian. For QA, the model predicts the answer option label, and outputs containing the full answer text are mapped back to the corresponding option during evaluation. For MR, the model is encouraged to generate intermediate reasoning, while only the final answer following the token Answer: is used for evaluation.

To balance training efficiency and mitigate forgetting, we also empirically set a replay size of 5,000 EN→UK MT examples and 5,000 GC examples from the first-stage datasets during second-stage training. Consequently, the second stage

Task	Zero-shot	After Stage 1	Improvements
MT (BLEU)	15.40	17.92	+2.5
GC (det / corr)	34.2 / 32.4	42.5 / 42.1	+8.3 / +9.7
SC (det / corr)	32.5 / 30.6	75.2 / 59.3	+42.7 / +28.8

Table 3: Effect of Stage-1 training on the Stage-1 tasks (development set).

jointly optimizes new tasks with replayed data to preserve prior capabilities while learning new ones.

5 Results and Discussion

Experimental setup. We compare four training strategies on the development set. **Baseline** directly evaluates the base Qwen3.5-2B-Instruct model without fine-tuning. **Joint** fine-tunes all six tasks jointly. **2-Stage-I** first trains on English→Ukrainian machine translation (EN→UK MT), spelling correction (SC), and grammar correction (GC), and then continues training on the full multi-task mixture with replayed first-stage data. **2-Stage-II** follows the same framework but postpones GC until the second stage. Unless otherwise stated, all results are reported on the development set. Our final submission adopts the 2-Stage-I configuration. Detailed hyperparameter settings are provided in Appendix A.

5.1 Results

Development-set results. We first evaluate the different training strategies on the development sets. Tables 2 show that ours outperforms over the zero-shot baseline across multiple tasks and all fine-tuned configurations improve substantially over the zero-shot baseline across tasks, confirming the value of task-specific fine-tuning.

Our final system (**2-Stage-I**) substantially improves Ukrainian generation in both EN→UK and CS→UK translation on both BLEU and chrF++. Notably, comparing the Stage-1-only results in Ta-

System	Machine Translation					QA	SC	GC	Math
	All chrF++	CS-UK BLEU	CS-UK chrF++	EN-UK BLEU	EN-UK chrF++	All EM	All EM	All EM	UkrMR EM
Baseline	18.46	5.38	25.44	0.55	11.47	36.33	8.38	2.18	20.00
TUMHN	16.08	3.93	19.77	1.01	12.39	40.13	16.50	6.80	4.40
Koshi	16.80	6.01	23.05	0.70	10.55	46.99	52.50	30.55	0.40
Ours	49.86	23.66	51.53	15.43	48.18	41.15	66.43	45.18	21.60

Table 4: Official test-set results on the Ukrainian track of the WMT 2026 Shared Task. For multi-subtask categories (QA, SC, GC), overall macro scores are reported here. Detailed sub-category breakdowns are provided in Appendix C.

ble 3 with the full two-stage results shows that the translation ability acquired in Stage 1 is not degraded by the later reasoning stage.

Among the training strategies, **2-Stage-I** provides the strongest overall balance across heterogeneous tasks, achieving the best performance on SC, QA, and MR while remaining competitive on GC.

Official test-set results. Table 4 reports the official test-set results of our final submission. Ours achieves the highest reported scores on all four tasks metrics. For question answering, our system achieves 41.15 EM, below Koshi’s 46.99 EM, but remains competitive with the other systems.

Compared with the official baseline, the largest improvements are observed on machine translation and correction tasks, demonstrating that staged multi-task fine-tuning effectively enhances Ukrainian generation and linguistic knowledge. More importantly, the results show that a single compact model can jointly support translation, correction, reasoning, and language understanding without sacrificing overall performance.

5.2 Training Strategy Analysis

Effect of staged-training Comparing Joint strategy and 2-Stage strategy as shown in Table 2, the 2-Stage method performs better on most tasks. Separating training into two stages lets the model first acquire translation and correction behaviors and only then take on reasoning, rather than trading them off from the start. We therefore adopt the staged setup as our main system.

Our training order follows a language-first approach. We first train the model on translation and correction tasks so that it can better handle Ukrainian before introducing QA and MR. The idea is that a stronger foundation in the target language may make it easier for the model to learn the later reasoning tasks. As shown in Table 3, Stage

1 improves MT, SC, and GC by 59.2% on average over the baseline. Stage 2 extends the model to reasoning and another translation direction. Importantly, most capabilities learned in Stage 1 are also preserved after Stage 2, suggesting that sequential fine-tuning can absorb new tasks without substantially overwriting prior knowledge.

Task composition and ordering. Given that we train in two stages, the composition and ordering of tasks within each stage may affect transfer and interference. Rather than randomly partitioning tasks, we explored different combinations and orderings and report the two configurations that performed best overall.

Moving GC from the first stage to the second (2-Stage-I vs. 2-Stage-II) changes both the correction and the downstream reasoning scores, showing that ordering and composition jointly determine which capabilities transfer and which interfere. Two contrasting effects illustrate this. The configuration with the strongest SC performance also achieves the strongest QA performance; since the SC data are derived from the QA corpus, this points to positive transfer between the two. Conversely, the degradation of GC after subsequent training reflects negative interference. Task composition should therefore account for task compatibility rather than treating all tasks as interchangeable.

Catastrophic forgetting and replay. To mitigate the GC degradation identified above, we retain a small replay buffer of representative Stage-1 examples and mix them into Stage-2. This allows the model to maintain training signals for previously learned tasks while introducing the new objectives, providing a simple mechanism for reducing capability drift.

We also use more English→Ukrainian than Czech→Ukrainian data to mitigate the MT degradation. Since Czech and Ukrainian are both Slavic

languages and share many lexical and morphological features, Czech→Ukrainian is easier to preserve, letting us spend more of the budget on the more forgetting-prone English→Ukrainian direction.

5.3 Output Analysis and Discussion

We qualitatively examined model outputs across all five tasks and group the observed behaviors into three types. Where relevant, we compare the final system (**2-Stage-I**), which is trained on the cleaned math data described in Section 3, with an ablation variant (**MR-Raw**) trained on the uncleaned, raw translated MR data, both are evaluated on the same test set.

(1) Repetition and degeneration. The most common failure is repetition, where the model repeats until it reaches the maximum length. This is rare on short inputs but frequent on longer ones. In MR task it occurs in only 5 of 125 low-difficulty items but in 69 of 125 medium-difficulty items. We also observe it in some of longer translation outputs. The problem is that a 2B model, adapted with a small number of parameters, may not always produce a stable stopping point once the output grows long and symbol-heavy. We clean the training data to mitigate it: the raw math solutions (**MR-Raw**) contained calculator markers such as « $48/2=24$ », a highly regular pattern that the model learned to repeat, and removing them in **2-Stage-I** reduces this. However, the remaining failures on hard math problems are driven by long derivations, and may be better addressed at decoding time, for example, by penalizing repetition.

(2) Format and prompt following. In general, the simpler tasks generally show strong format compliance. However, a small fraction of correction outputs only contain a dash (-)(around 6% for spelling and 8% for grammar). These are often associated with inputs beginning with a dash or bullet, suggesting that the model sometimes copies salient surface tokens from the input rather than producing the requested structured field.

The largest format failures occur in MR. Although low-difficulty instances usually produce directly usable answers, 118 of 250 predictions (47.2%) do not contain the final numerical answer. This partly results from our prompt, which explicitly encourages step-by-step reasoning. While this may benefit mathematical reasoning, it also lengthens generation and can exhaust the token budget

before a final answer is produced, especially on harder problems. Combined with the repetition issues discussed above, this creates a trade-off between encouraging reasoning and producing an easily extractable answer.

(3) Reasoning language and cross-lingual consistency. We further examined whether the model maintains Ukrainian when generating explanations, given that the task prompts are written in English. Overall, we find little evidence of systematic cross-lingual drift. The natural-language portions of MR outputs are predominantly Ukrainian, including the longer reasoning traces. Apparent Latin-script content is mostly mathematical notation rather than English prose, and Russian-specific characters appear only rarely in MT outputs.

The remaining language errors are not switches from Ukrainian to English or Russian, but local contamination from degenerate generations, such as repeated punctuation, symbols, or emojis. Language consistency is thus not the main bottleneck; improving generation stability and format control is likely to have greater impact.

6 Conclusion

We presented our submission to the WMT 2026 Shared Task on LLMs with Limited Resources for Ukrainian. Our system fine-tunes a single Qwen3.5-2B-Instruct model using a two-stage multi-task training strategy with replay to mitigate catastrophic forgetting.

Development-set experiments show that staged training provides a better balance across heterogeneous tasks than straightforward joint optimization. In particular, we observe positive transfer from spelling correction to question answering, while grammar correction remains the task most vulnerable to catastrophic forgetting.

On the official test set, our system achieves the highest reported scores on all evaluated tasks except QA. These results demonstrate that carefully scheduled multi-task fine-tuning enables compact multilingual LLMs to effectively balance diverse tasks under limited-resource constraints.

Acknowledgements

KH was also a co-organiser of the WMT26 Shared Task on Multitask LLMs with Limited Resources. The implementation and experiments for the system were all carried out by ZF and LY, as part of a course project advised by KH. All information

shared with ZF and LY for this work was also publicly available to the other participants of the shared task.

This work was co-funded by the European Union (ERC, EPICAL, 101141712). Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Research Council. Neither the European Union nor the granting authority can be held responsible for them.

References

- Dmytro Chaplinskyi. 2023. [Introducing UberText 2.0: A corpus of modern Ukrainian at scale](#). In *Proceedings of the Second Ukrainian Natural Language Processing Workshop*, pages 1–10, Dubrovnik, Croatia. Association for Computational Linguistics.
- Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Alex Castro-Ros, Marie Pellat, Kevin Robinson, and 16 others. 2022. [Scaling instruction-finetuned language models](#). *Preprint*, arXiv:2210.11416.
- Karl Cobbe et al. 2021. Training verifiers to solve math word problems. *arXiv:2110.14168*.
- Edouard Grave, Piotr Bojanowski, Prakhara Gupta, Armand Joulin, and Tomas Mikolov. 2018. Learning word vectors for 157 languages. In *Proceedings of LREC*.
- Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2021a. Measuring massive multitask language understanding. In *International Conference on Learning Representations (ICLR)*.
- Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. 2021b. [Measuring mathematical problem solving with the MATH dataset](#). In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*, volume 1.
- Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*.
- Yun Luo, Zhen Yang, Fandong Meng, Yafu Li, Jie Zhou, and Yue Zhang. 2025. [An empirical study of catastrophic forgetting in large language models during continual fine-tuning](#). *Preprint*, arXiv:2308.08747.
- Shu Okabe, Daryna Dementieva, Marion Di Marco, Lukas Edman, Katharina Haemmerl, Marko Měškank, Anita Hendrichowa, and Alexander Fraser. 2025. [Findings of the WMT 2025 shared task LLMs with limited resources for Slavic languages: MT and QA](#). In *Proceedings of the Tenth Conference on Machine Translation*, pages 503–519, Suzhou, China. Association for Computational Linguistics.
- Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. [BLEU: a method for automatic evaluation of machine translation](#). In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.
- Martin Popel, Lucie Poláková, Michal Novák, Jindřich Helcl, Jindřich Libovický, Pavel Straňák, Tomáš Krabáč, Jaroslava Hlaváčová, Mariia Anisimova, and Tereza Chláňová. 2024. [Charles translator: A machine translation system between ukrainian and czech](#). *Preprint*, arXiv:2404.06964.
- Maja Popović. 2017. [chrf++: words helping character n-grams](#). In *Proceedings of the Second Conference on Machine Translation*, pages 612–618, Copenhagen, Denmark. Association for Computational Linguistics.
- Matt Post. 2018. [A call for clarity in reporting BLEU scores](#). In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 186–191, Brussels, Belgium. Association for Computational Linguistics.
- Mariana Romanyshyn, Oleksiy Syvokon, and Roman Kyslyi. 2024. [The UNLP 2024 shared task on fine-tuning large language models for Ukrainian](#). In *Proceedings of the Third Ukrainian Natural Language Processing Workshop (UNLP) @ LREC-COLING 2024*, pages 67–74, Torino, Italia. ELRA and ICCL.
- 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](#). *Preprint*, arXiv:2509.22490.
- Mykyta Syromiatnikov and NLPForUA team. 2025. [Dumy: Ukrainian multidomain reasoning dataset](#).
- Oleksiy Syvokon and Mariana Romanyshyn. 2024. Creating a large benchmark for ukrainian text correction. In *Proceedings of the Third Ukrainian Natural Language Processing Workshop (UNLP)*.
- NLLB Team, Marta R. Costa-jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Hefernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, Anna Sun, Skyler Wang, Guillaume Wenzek, Al Youngblood, Bapi Akula, Loic Barrault, Gabriel Mejia Gonzalez, Prangthip Hansanti, and 20 others. 2022. [No language left behind: Scaling human-centered machine translation](#). *Preprint*, arXiv:2207.04672.

Jörg Tiedemann. 2012. [Parallel data, tools and interfaces in OPUS](#). In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC’12)*, pages 2214–2218, Istanbul, Turkey. European Language Resources Association (ELRA).

Liang Wang, Nan Yang, Xiaolong Huang, Linjun Yang, Rangan Majumder, and Furu Wei. 2024. [Multilingual e5 text embeddings: A technical report](#). *Preprint*, arXiv:2402.05672.

Hanna Yukhymenko, Anton Alexandrov, and Martin Vechev. 2026. [Recovered in translation: Efficient pipeline for automated translation of benchmarks and datasets](#). In *Findings of the Association for Computational Linguistics: ACL 2026*, San Diego, California, United States.

A Hyperparameters & Training Specs

In this section, we provide the hyperparameters used for both model training and decoding.

Hyperparameter	Value
Optimizer	AdamW
Learning Rate	2×10^{-4}
Learning Rate Schedule	Cosine decay with 3% warmup
Precision	bfloat16
Batch	4×8
Epoch	5
Max Sequence Length	3072
Rank	16
Alpha	32
Target Modules	q_proj, k_proj, v_proj, o_proj, gate_proj up_proj, down_proj
LoRA Dropout	0.05

Table 5: Training and decoding hyperparameter configurations.

B Evaluation Metrics and Settings

To ensure reproducibility, we report the exact tools and parameter settings used for evaluating Machine Translation. Both metrics are computed at the corpus level using the sacrebleu library (Post, 2018):

- **BLEU** (Papineni et al., 2002): We compute corpus-level BLEU using sacrebleu.corpus_bleu with the default SacreBLEU settings, including the default 13a tokenizer.
- **chrF++** (Popović, 2017): We compute corpus-level chrF++ using sacrebleu.corpus_chrf with word_order=2, which incorporates word n -gram information in addition to character

n -grams. All other parameters are left at their SacreBLEU default values.

C Detailed Test-Set Performance Breakdowns

Table 6 and Table 7 report the detailed sub-metric evaluations on official downstream tasks for the WMT 2026 Shared Task (Ukrainian track).

System	Question Answering		Math
	UkrQA EM	UkrMMLU EM	UkrMR EM
Baseline	30.76	41.90	20.00
TUMHN	35.82	44.44	4.40
Koshi	45.01	48.97	0.40
Ours	37.42	44.88	21.60

Table 6: Detailed test-set results on Question Answering and Math reasoning.

System	Spelling Correction		Grammar Correction	
	Wrong EM	Corrected EM	Wrong EM	Corrected EM
Baseline	8.55	8.20	2.70	1.65
TUMHN	22.20	10.80	10.35	3.25
Koshi	60.10	44.90	36.55	24.55
Ours	71.70	61.15	45.25	45.10

Table 7: Detailed test-set results on Spelling and Grammar Correction.