Yandex Submission to the WMT25 General Translation Task

Nikolay Karpachev Ekaterina Enikeeva Dmitry Popov
Arsenii Bulgakov Daniil Panteleev Dmitrii Ulianov Artem Kryukov Artem Mekhraliev

Yandex

Abstract

This paper describes Yandex submission to the WMT25 General Machine Translation task. We participate in English-to-Russian translation direction and propose a purely LLM-based translation model. Our training procedure comprises a training pipeline of several stages built upon YandexGPT, an in-house general-purpose LLM. In particular, firstly, we employ continual pretraining (post-pretrain) for MT task for initial adaptation to multilinguality and translation. Subsequently, we use SFT on parallel document-level corpus in the form of P-Tuning. Following SFT, we propose a novel alignment scheme of two stages, the first one being a curriculum learning with difficulty schedule and a second one - training the model for tag preservation and error correction with human post-edits as training samples. Our model achieves results comparable to human reference translations on multiple domains.

1 Introduction

We participate in the WMT25 General Machine Translation task and propose a purely LLM-based translation system.

Large Language Models (LLMs) have recently redefined the state-of-the-art in machine translation, demonstrating strong capabilities that yield near-human quality outputs on vast collection of language pairs. Their performance has consistently surpassed that of the previous generation of specialized Neural Machine Translation (NMT) systems, marking a significant paradigm shift in the field. The recent WMT24 General Machine Translation contest provides compelling evidence of this trend, where top-performing systems, predominantly based on LLMs, achieved translation scores remarkably close to human reference translations.

Still, the human parity claim remains disputable and thorough evaluations have shown that for a variety of high-resource morphologically rich languages, although quite high, the performance of LLMs still lags behind professional human translations. All state-of-the-art LLM systems exhibit a noticeable pattern of literal translations with fluency and naturalness of generations significantly worse than that of native human translations.

In this work, we propose a novel pipeline for LLM adaptation to the MT task, building upon our previous year submission (Elshin et al., 2024). The main goal of this work is to explore tools and techniques for improving the performance of an already capable MT-specific LLM model with near human performance.

The system comprises a pipeline of several adaptation stages built upon 7-billion YandexGPT, an inhouse proprietary general-purpose language model.

- First, we employ continual pretraining for robust adaptation of general-purpose pretrained model to the task of translation and multilinguality (post-pretrain)
- Following post-pretrain, the model is finetuned on a cleaner corpus of automatically collected books translations
- Subsequently, the system undergoes an alignment procedure consisting of two primary stages
  - Contrastive Preference Optimization (CPO) with curriculum learning for low-resource document-level adaptation, targeted at fluency and cohesion improvement
  - Second alignment stage focused on tackling the model shortcomings and tag preservation training

The resulting model (Yandex) was subsequently fine-tuned on the WMT dataset, producing the Yandex+WMT model which constitutes our submis-

Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 293–305
November 8–9, 2025.

sion to the WMT25 General Machine Translation task.

For the English-to-Russian translation direction our resulting system is significantly better than all of the previous year contenders and achieves results comparable to major foundational LLMs on this year's benchmark.

2 System Overview

In this section we describe the training pipeline of the system and details of the inference procedure. We also provide automatic metrics and human evaluation results for the key components of the system.

2.1 Pretrain

The base model that we use is a 7 billion parameter version of YandexGPT, an in-house general-purpose large language model. It is a decoder-only model of an architecture similar to Touvron et al. (2023) trained on a collection of data primarily consisting of Russian and English texts.

In our experiments we use the pre-train stage of YandexGPT as the starting point for machine translation specific fine-tuning.

2.2 Post-pretrain

As an initial adaptation for multilinguality and translation task, we perform a continual pretraining using recipe similar to Alves et al. (2024).

We fine-tune the model using full weights fine-tuning with a standard cross-entropy loss on a mixture of pretrain dataset and MT data with a ratio of translation data of 30%.

The parallel translation data is collected using a matching pipeline similar to Bitextor (Esplà-Gomis, 2009). It involves matching multilingual websites as candidates for parallel documents, followed by a series of alignment and filtering steps (Thompson (2019), Artetxe and Schwenk (2019)).

For each example of parallel translation data, we construct two samples for continual pretraining via concatenation in both ordering variants: english text concatenated with russian translation via two new-line separators and vice-versa.

In our experiments we have observed that the optimal ratio of incorporating MT data is around 30% and it is beneficial to mirror all en-ru training samples in reverse direction.

2.3 Supervised Fine-tuning

Following post-pretrain, we employ supervised fine-tuning (SFT) in order to focus the model solely on the translation task and enforce more precise outputs without hallucinations.

We use an in-house dataset of parallel English and Russian books aligned at the paragraph-level. We apply filtering by length and train on fragments with maximum length of 1k sentence-piece tokens (Kudo and Richardson, 2018). In addition to that, we only use paragraph pairs with the same number of sentences in the source and translation text.

In terms of the training procedure, we have experimented with both standard SFT finetuning and sparse methods like LoRa (Hu et al., 2021) and PTune variants (Liu et al. (2021b), Liu et al. (2021a)).

Our experiments have shown that not only does not the Full Finetuning improve quality upon parameter-efficient training strategies but it also leads to the quality degradation after subsequent alignment. We hypothesize that the root cause for this phenomenon is the "knowledge forgetting". Specifically, more extensive fine-tuning methods make SFT checkpoint less sensitive to the pre-trained LLM knowledge and hence over-optimize for the parallel dataset during the SFT stage.

The resulting system involves training with P-Tuning v2 (Liu et al., 2021a) with two trainable P-Tuning blocks each having the size of 100 ptune tokens placed

- At the start of the input string, before the English source
- Between English source and Russian translation to be generated

In Table 1 we report automatic evaluation results of models from pretrain and SFT stages. We measure MetricX-24 (Juraska et al., 2024) in both reference-based and reference-less QE variants as well as fluency score, which is an in-house monolingual classifier that measures the grammatical and lexical correctness of the Russian translation. We use the same subset of WMT24 test data as in human evaluations for results consistency.

2.4 First-Stage Alignment

In this section we describe the key components of the subsequent step in the training pipeline - the first stage of alignment fine-tuning.

The primary goal of alignment is to effectively make use of another form of training signal - user

Model	METRICX-24-XXL	METRICX-24-XXL-QE	Fluency
YandexGPT Pretrain	4.746	4.262	0.754
MT Postpretrain	4.899	4.499	0.772
SFT (PTune)	4.272	3.854	0.795

Table 1: Comparison of model performance on WMT24 testset.

preferences data or algorithmic rankings of translations with varying quality. In contrast to SFT, alignment techniques like reinforcement learning or contrastive learning allow to perform not only "likelihood" traning on good reference, but also "unlikelihood" training on the data with proven deficiencies or more sophisticated ranking-based approaches.

2.4.1 Sentence-Level Data

The first portion of the training dataset is an internal collection of historic human evaluations of various model generations. This dataset consists of sentence triplets ("source", "winner_translation", "loser_translation") and has a size of several tens of thousand examples.

Namely, "winner_translation" is a preferred hypothesis, and "loser_translation" is a dispreferred. All rankings were done by professional human annotators via a platform similar to Amazon Mechanical Turk.

The samples themselves are single sentences of varying length, typically no more than 100 tokens.

2.4.2 Document-Level Data

As training solely on sentences would not expose the model to complex and practically challenging discourse phenomena of translation like deixis, ellipsis and lexical cohesion we also collected several specific datasets to emphasize such phenomena during training.

The first one is specifically targeted on fluency and coherence improvement (**fluency repair**):

1. Generate paragraph translations with the sentence-level translation model.

2. "Improve" the non-coherent paragraph translations using a monolingual general-purpose LLM.

3. Train on such fluency corrections using contrastive learning, wherein the "positive" hypothesis would be the smoothed and fluent translation and the "negative" hypothesis would be the original sentence-wise translation.

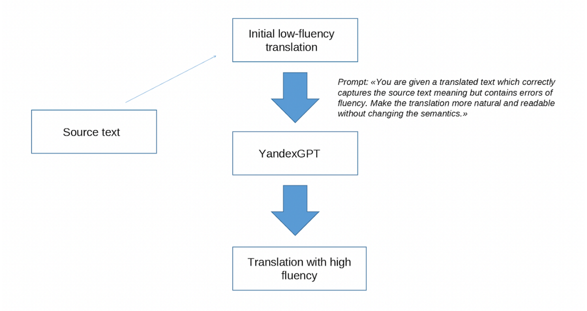
Initial low-fluency translation
Prompt: «You are given a translated text which correctly captures the source text meaning but contains errors of fluency. Make the translation more natural and readable without changing the semantics.»
Source text
YandexGPT
Translation with high fluency

Figure 1: Fluency repair procedure.

We also collect several thousands of side-by-side comparisons of different model generations on paragraph-level source data.

In addition to the contrastive learning triplet data, we found it beneficial to mix triplet data with a small high-quality SFT set of manually written translations created by experts with high language proficiency. We add several thousand such samples to the alignment stage.

2.4.3 Curriculum Learning

In the previous two sections we have outlined the data collection procedure of two parts: sentence-level and document-level data. Sentence-level dataset is significantly larger, consisting of more than 100.000 samples, while the whole document or paragraph-level translations corpus is almost an order less.

This leads to a data imbalance issue during training. Uniform mixture of both sentence- and document-level sources would be highly skewed towards sentence part and in our experiments it produced results highly similar to training only on sentences. Upsampling of document-level data would result in overfitting.

This outlined problem could be handled through training with cirruculum learning (Bengio et al., 2009). We employ a difficulty schedule, first training solely on sentence corpus and then switching to documents at the end of the training.

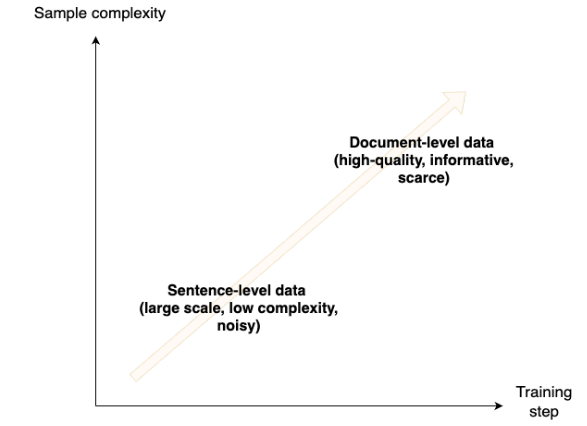
Sample complexity
Document-level data (high-quality, informative, scarce)
Sentence-level data (large scale, low complexity, noisy)
Training step

Figure 2: Curriculum learning with sentences-to-documents adaptation.

2.4.4 Training Procedure

We train using Contrastive Preference Optimization (CPO) objective (Xu et al., 2024)

$$\mathcal{L}(\pi_\theta) = -\mathbb{E}_{(x,y_w,y_l)\sim\mathcal{D}}\Bigg[\log\sigma\Bigg(\beta\log\pi_\theta(y_w|x) - \beta\log\pi_\theta(y_l|x)\Bigg)\Bigg]$$

with a cross-entropy regularizer:

$$\min_\theta \underbrace{\mathcal{L}(\pi_\theta, U)}_{\mathcal{L}_{prefer}} - \underbrace{\mathbb{E}_{(x,y_w)\sim\mathcal{D}}[\log\pi_\theta(y_w|x)]}_{\mathcal{L}_{NLL}}.$$

We have observed that using higher weights before regularizer cross-entropy term leads to more literal and adequacy-boosting translations (close to the SFT fine-tuning), while giving more weight to the contrastive learning term makes the model much more fluent, but prone to hallucinations.

Overall, the training objective for contrastive triplets is CPO with cross-entropy regulatization weight set to 0.1:

$$\min_\theta \underbrace{\mathcal{L}(\pi_\theta, U)}_{\mathcal{L}_{prefer}} - 0.1 \cdot \underbrace{\mathbb{E}_{(x,y_w)\sim\mathcal{D}}[\log\pi_\theta(y_w|x)]}_{\mathcal{L}_{NLL}}.$$

For the high-quality references, we use standard SFT with learning rate 20 times higher than those for contrastive samples.

We train with one epoch and follow a triangular learning rate schedule with the warmup of 10% of steps and linear decay.

Sentence- and document-level portions of the data are shuffled and the document-level data consists of fluency repair and side-by-side triplets mixed with SFT samples for high-quality references.

2.5 Second-Stage Alignment

During the second stage of alignment, our primary goal is to precisely address specific model deficiencies while developing structure-preserving capabilities. This phase focuses on fine-grained tuning of the model through targeted interventions.

We concentrate on two parallel objectives: first, we aim to fix the common errors of the model after the initial alignment phase. Second, we enhance the model's ability to maintain specific structural elements of the input data (such as HTML tags, list orderings, dialogue formatting etc.).

In practice, our approach involves collecting on-policy data that captures examples of typical errors of the model from the first alignment stage. We then perform post-editing of those on-policy translations, forming a contrastive dataset where post-edits serve as positive examples and the original model outputs as negative ones. We also perform an additional adaptation for WMT topics and construct a dataset derived from WMT24 data annotated with the RATE protocol (Popov et al., 2025). We create contrastive pairs by selecting translations where the average of fluency and accuracy scores differs by more than 5 points, as detailed in Section 2.5.1. The structure preservation dataset, also described in Section 2.5.1, similarly features targeted differences between negative and positive examples, focusing on maintaining specific structural elements rather than completely rewriting translations.

2.5.1 Datasets

Human post-editing

Following active learning paradigm, we collect human-written post-edits of first-stage alignment model generations. This naturally results in triplets of ("source text", "model translation", "human post-edit"). Hereby, the triplets contain a more concentrated signal that specifically targets model error corrections.

On-policy side-by-side comparisons

In addition to human-written post-edits, we collect side-by-side comparisons of on-policy model generations. The generations are obtained via standard sampling with temperature of 1.0 and side-by-side evaluation is conducted by professional annotators.

The impact of these two portions of the data can be formalized as

(a) eliminating systematic bias of the model in the form of error correction done by human

(b) decreasing variance of the model by training on on-policy comparisons

Structure preserving training

In order to translate structured data, such as HTML web pages, documents with formulae blocks, or subtitles with clear replics borders, we specifically train the model for tagged data translation.

Following Elshin et al. (2024), we convert all tags or separators to the universal tag ({ for the opening paired tag and } for the closing paired tag, each unpaired tag is converted to {})

Consequently, each input containing tags would first be converted to the universal tag format, for example, "<title>Paper Index - EMNLP 2021 Sixth Conference on Machine Translation (WMT21)</title>" would correspond to "{Paper Index - EMNLP 2021 Sixth Conference on Machine Translation (WMT21)}" input during inference.

This implies that for the correct translation of tagged data, the model should be able to preserve the exact bracket sequence given at the input, the number and the sequence of brackets.

We explicitly train for this property using a rule-based reward; for each (source, translation) pair it is algorithmically possible to determine whether the given translation has correctly preserved the tag sequence. Hence, one can construct training samples in an unsupervised way using diverse decoding or beam search:

1. Sample a set of model generations using diverse decoding or wide beam search
2. Score all outputs using rule-based reward that verifies the tag preservation
3. If possible, create a contrastive triplet wherein the positive example contains the output with correctly preserved tags and the negative one contains tag errors

Aiming to keep only informative samples, we select only the sources with tag error in the top-1 model hypothesis and take the most probable example with correctly translated tags to maximize the general translation quality.

WMT24 data

For domain adaptation purposes, we leverage a dataset annotated using the RATE protocol (Popov et al., 2025). This dataset encompasses all documents from WMT24 General Translation Task along with translations from 8 systems participating in the contest, comprising approximately 4,000 segments. The RATE protocol provides detailed information about error spans as well as pointwise accuracy and fluency scores on a 100-point scale.

To generate a contrastive training set from this resource, we take all translation pairs and select only those where the average of fluency and accuracy scores differs by more than 5 points. This selection process yields a dataset containing slightly over 7,000 contrastive examples. It's worth noting that the resulting dataset contains triplets that share the same source text, and, in some cases, a system translation may serve as a positive example in one triplet while appearing as a negative example in another, depending on the quality of alternatives it's being compared against.

2.6 Decoding

We employ a mixed decoding strategy by merging paragraph sequences into larger decoding chunks, hereby decoding with a **local context**.

Inference with Local Context

It is clear that an accurate translation of the document should be done with its full context. However, in practice we have observed that current translation models exhibit inferior performance when given inputs of sufficiently large size.

Given that translation quality starts to decline from several hundred tokens, we propose a hybrid decoding strategy.

1. Set decoding block size to 100 tokens.
2. For a given document, merge sequential paragraphs into blocks greedily, while the currently accumulated block size is less than the decoding block size.
3. Consider the current sequence of paragraphs a single decoding "block".
4. Next blocks are constructed accordingly.

Model	METRICX-24-XXL	METRICX-24-XXL-QE	Fluency
SFT (PTune)	4.272	3.854	0.795
CPO Stage 1	2.417	2.167	0.902
DPO Stage 2	2.369	2.136	0.895
DPO Stage 2 + tags	2.406	2.192	0.898
DPO Stage 2 + tags + WMT	2.253	2.171	0.911

Table 2: Model quality dynamics through alignment stages (WMT24 testset).

	WMT24		WMT25	
domain	segments	documents	segments	documents
literary	63	7	6	2
news	94	16	41	14
social	272	33	27	9
speech	66	66	62	62

Table 3: Testsets descriptive statistics.

To preserve the paragraph structure, we wrap each paragraph in the block with { } tags, thus making sure that the translated document would contain the same number of paragraphs as the source document.

Table 2 shows the quality dynamic of alignment components on WMT24 testset. The first alignment stage displays a significant improvement over SFT with a large margin on all metrics, whereas subsequent alignment stages do not bring statistical improvements in MetricX. We hypothesize that automatic evaluation becomes insensitive from a certain quality of translations and does not capture subtle differences that human annotators still observe.

3 Human Evaluation Results

Due to constraints on time and human resources we selected subsets from two testsets for human evaluation: 495 segments from WMT24 general MT testset and 136 segments from WMT25 general MT blindset. These subsets are hereafter referred to as WMT24 and WMT25, respectively. Obviously, the human translations of WMT25 testset could not be included in this evaluation campaign since they have not been publicly released yet. The number of segments and their distribution across domains is presented in Table 3.

Two protocols were implemented to evaluate MT quality of our submission: the ESA protocol (Kocmi et al., 2024), following WMT guidelines, and the RATE protocol, introduced in Popov et al. (2025). The annotators' qualifications and detailed annotation setup are described in Appendix A.

Yandex+WMT is compared to Yandex model (referred to as DPO Stage 2 + tags above) and several LLM translations. Claude3.7 and GPT-4 translations for WMT24 testset were obtained directly from WMT24 publicly released data. The WMT25 testset was translated using Claude 4 and GPT-4.1 with a simple prompt *"You are a professional English-to-Russian translator. Your goal is to accurately convey the meaning and nuances of the original English text while adhering to Russian grammar, vocabulary, and cultural sensitivities. Produce only the Russian translation, without any additional explanations or commentary. Translate the following text: {input text}"*.

We report segment-level ESA scores alongside error counts and MQM-like scores calculated as $5 \times major + minor$ in Table 4. For the RATE protocol, separate scores for accuracy, fluency, and style are reported, as well as error category statistics. We also report error-per-token statistics and macro-averaged counts by document, domain, or both in Appendix B. Following the WMT methodology, we compute pairwise statistical significance of the differences by Wilcoxon signed rank test and group systems into clusters represented numerically in the tables. Both evaluation methods on WMT25 testset confirm that Yandex+WMT outperforms Yandex and demonstrates statistically significant improvement over the compared LLMs on both testsets. RATE results provide more interpretable differentiation between Yandex and Yandex+WMT: the fluency score shows a measurable increase, while

	segment level					counts per token		
system	errors	major	minor	MQM	ESA	errors	major	minor
Claude 3.7	2.35_3	0.79_2	1.56_3	5.65_5	79.48_3	0.09	0.03	0.06
GPT-4.1	2.64_4	1.10_3	1.54_3	7.34_3	75.41_5	0.11	0.05	0.06
RefA	2.20_2	1.10_3	1.10_2	7.44_4	78.38_4	0.10	0.05	0.05
Yandex	1.51_1	0.74_1	0.76_1	5.06_1	82.16_1	0.07	0.03	0.04
Yandex+WMT	1.50_1	0.80_2	0.70_1	5.35_2	81.30_2	0.07	0.04	0.03

Table 4: Segment-level ESA annotation results on WMT24 testset.

	segment level					counts per token		
system	errors	major	minor	MQM	ESA	errors	major	minor
Claude 4	5.94_3	2.77_3	3.18_4	17.22_3	68.05_4	0.06	0.03	0.03
GPT-4.1	4.67_2	2.56_2	2.11_3	15.69_2	71.77_3	0.05	0.03	0.02
Yandex	4.02_1	2.42_1	1.60_2	15.51_1	72.19_2	0.04	0.03	0.02
Yandex+WMT	3.69_1	2.37_1	1.32_1	15.46_1	73.51_1	0.04	0.02	0.01

Table 5: Segment-level ESA annotation results on WMT25 testset.

differences in accuracy remain statistically non-significant.

4 Conclusion

In this paper, we describe Yandex submission to the WMT25 General Translation task. For English-to-Russian translation direction, our model outperforms all systems from WMT24 competition and achieves results comparable to major foundational LLMs on WMT25 benchmark, as measured by ESA and RATE human evaluation protocols. According to the official human evaluation results our model achieves parity with human reference translations on ESA score.

We present a detailed description of our training procedure as well as giving the rationale for different training steps. Our pipeline includes multi-stage alignment procedure specifically designed to improve the quality of an already capable machine translation system, with the performance close to the human one.

We employ novel techniques, such as curriculum learning with sentences-to-documents adaptation, training on human post-edits and fine-tuning for tagged data using rule-based reward. Our results, measured both in automatic metrics and human evaluations, demonstrate the effectiveness of the proposed pipeline as well as high overall translation quality of the system.

	segment level							
system	errors	major	minor	MQM	accuracy	fluency	style	RATE
Claude 4	13.81_3	2.78_2	10.03_4	23.96_2	70.78_2	58.90_4	89.06_3	23.81_3
GPT-4.1	10.94_2	2.08_1	7.86_3	18.38_1	73.69_1	69.04_3	92.04_2	18.33_2
Yandex	9.07_1	2.21_1	5.68_2	17.77_1	70.42_2	74.39_2	92.94_2	16.95_1f
Yandex+WMT	8.76_1	2.08_1	5.00_1	16.93_1	68.64_3	77.46_1	94.82_1	16.11_1

Table 6: Segment-level RATE annotation results on WMT25 testset.

References

Duarte M. Alves, José P. Pombal, Nuno M. Guerreiro, Pedro Henrique Martins, João Alves, Amin Farajian, Ben Peters, Ricardo Rei, Patrick Fernandes, Sweta Agrawal, Pierre Colombo, José G. C. de Souza, and André Martins. 2024. Tower: An open multilingual large language model for translation-related tasks. *ArXiv*, abs/2402.17733.

Mikel Artetxe and Holger Schwenk. 2019. Massively multilingual sentence embeddings for zero-shot cross-lingual transfer and beyond. *Transactions of the Association for Computational Linguistics*, 7:597–610.

Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. 2009. Curriculum learning. In *International Conference on Machine Learning*.

Denis Elshin, Nikolay Karpachev, Boris Gruzdev, Ilya Golovanov, Georgy Ivanov, Alexander Antonov, Nickolay Skachkov, Ekaterina Latypova, Vladimir Layner, Ekaterina Enikeeva, Dmitry Popov, Anton Chekashev, Vladislav Negodin, Vera Frantsuzova, Alexander Chernyshev, and Kirill Denisov. 2024. From general llm to translation: How we dramatically improve translation quality using human evaluation data for llm finetuning. In *Conference on Machine Translation*.

Miquel Esplà-Gomis. 2009. Bitextor: a free/open-source software to harvest translation memories from multilingual websites. In *Beyond Translation Memories: New Tools for Translators Workshop*, Ottawa, Canada.

J. Edward Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *ArXiv*, abs/2106.09685.

Juraj Juraska, Daniel Deutsch, Mara Finkelstein, and Markus Freitag. 2024. Metricx-24: The google submission to the wmt 2024 metrics shared task. pages 492–504.

Tom Kocmi, Vilém Zouhar, Eleftherios Avramidis, Roman Grundkiewicz, Marzena Karpinska, Maja Popović, Mrinmaya Sachan, and Mariya Shmatova. 2024. Error span annotation: A balanced approach for human evaluation of machine translation. In *Proceedings of the Ninth Conference on Machine Translation*, pages 1440–1453, Miami, Florida, USA. Association for Computational Linguistics.

Taku Kudo and John Richardson. 2018. SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 66–71, Brussels, Belgium. Association for Computational Linguistics.

Xiao Liu, Kaixuan Ji, Yicheng Fu, Zhengxiao Du, Zhilin Yang, and Jie Tang. 2021a. P-tuning v2: Prompt tuning can be comparable to fine-tuning universally across scales and tasks. *ArXiv*, abs/2110.07602.

Xiao Liu, Yanan Zheng, Zhengxiao Du, Ming Ding, Yujie Qian, Zhilin Yang, and Jie Tang. 2021b. Gpt understands, too. *AI Open*, 5:208–215.

Dmitry Popov, Vladislav Negodin, Ekaterina Enikeeva, Iana Matrosova, Nikolay Karpachev, and Max Ryabinin. 2025. Refined assessment for translation evaluation: Rethinking machine translation evaluation in the era of human-level systems. Under review.

Brian Thompson. 2019. Vecalign: Improved sentence alignment in linear time and space. In *Conference on Empirical Methods in Natural Language Processing*.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aur'elien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023. Llama: Open and efficient foundation language models. *ArXiv*, abs/2302.13971.

Haoran Xu, Amr Sharaf, Yunmo Chen, Weiting Tan, Lingfeng Shen, Benjamin Van Durme, Kenton Murray, and Young Jin Kim. 2024. Contrastive preference optimization: Pushing the boundaries of llm performance in machine translation. *ArXiv*, abs/2401.08417.

domain	system	errors	major	minor	MQM	ESA
literary	**Claude 4**	2.39	**0.78**	1.61	**5.50**	82.78
	GPT-4.1	2.72	0.89	1.83	6.56	75.78
	Yandex	**1.67**	1.00	**0.67**	7.89	**87.56**
	Yandex+WMT	2.67	1.72	0.94	10.67	77.67
news	**Claude 4**	4.15	2.04	2.11	12.35	72.76
	GPT-4.1	3.50	1.83	1.67	11.19	74.67
	Yandex	3.34	1.96	1.38	12.24	73.49
	Yandex+WMT	**2.38**	**1.58**	**0.80**	**10.01**	**79.65**
social	**Claude 4**	5.93	2.47	3.46	15.86	68.00
	GPT-4.1	4.25	2.06	2.19	13.11	73.51
	Yandex	**3.38**	**2.02**	**1.36**	**12.17**	**73.95**
	Yandex+WMT	3.80	2.42	1.38	15.53	71.77
speech	**Claude 4**	7.48	3.57	3.91	22.17	63.53
	GPT-4.1	5.81	3.42	2.39	20.68	68.72
	Yandex	4.97	3.03	1.95	19.87	69.07
	Yandex+WMT	**4.61**	**2.94**	**1.68**	**19.49**	**69.82**

Table 7: WMT25 ESA annotation results - segment-level average by domains.

A Annotation details

The evaluation was conducted by two distinct groups of annotators:

1. ESA experts: an in-house group of Russian language natives who successfully passed the C1-level English test and regularly participate in translation evaluation campaigns. These experts received training to annotate translations according to ESA instructions. Quality control was maintained through manually prepared golden annotations.

2. RATE experts: a smaller in-house group of Russian language natives with qualification in Linguistics or Translation who underwent a multi-step selection process based on translation, post-editing and fact-checking competencies. As shown in Popov et al. (2025), this rigorous selection procedure may be crucial for ensuring annotation quality when evaluating high-quality translations.

B Extended human evaluation results

B.1 WMT25 ESA results

system	errors	major	minor	MQM	ESA
Claude 4	4.99	2.21	2.77	13.97	71.77
GPT-4.1	4.07	**2.05**	2.02	12.88	73.17
Yandex	**3.34**	**2.00**	1.34	**13.04**	**76.02**
Yandex+WMT	**3.37**	**2.16**	**1.20**	**13.93**	74.72

Table 8: WMT25 ESA annotation results - segment-level values macro averaged by domains.

system	errors	major	minor	MQM	ESA
Claude 4	6.66	3.15	3.52	19.55	65.90
GPT-4.1	5.20	2.96	2.24	18.03	70.34
Yandex	4.47	2.70	1.77	17.56	70.69
Yandex+WMT	**4.12**	**2.63**	**1.49**	**17.33**	**71.81**

Table 9: WMT25 ESA annotation results - document-level scores.

system	errors	major	minor	MQM	ESA
Claude 4	4.98	2.22	2.76	13.97	71.73
GPT-4.1	4.06	2.05	2.02	12.86	73.17
Yandex	**3.34**	**2.00**	1.34	**13.03**	**75.99**
Yandex+WMT	**3.36**	2.16	**1.21**	13.89	74.77

Table 10: WMT25 ESA annotation results - document-level scores macro-averaged by domain.

domain	system	errors	major	minor	MQM	accuracy	fluency	style	RATE
literary	**Claude 4**	8.83	1.42	6.33	13.42	80.08	68.58	94.17	13.33
	GPT-4.1	8.83	**0.67**	7.17	10.50	86.42	74.33	**95.33**	11.50
	Yandex	**5.08**	0.92	**3.75**	**8.33**	80.08	**82.00**	94.58	**8.75**
	Yandex+WMT	6.58	0.83	4.25	8.83	**83.58**	80.92	94.50	9.50
news	**Claude 4**	10.41	2.09	7.52	18.01	70.09	67.20	88.57	17.72
	GPT-4.1	7.91	1.61	5.48	13.65	**76.39**	75.10	93.66	13.61
	Yandex	**6.37**	1.72	**3.77**	12.91	73.83	79.57	93.29	11.83
	Yandex+WMT	6.84	**1.39**	3.83	**12.55**	72.33	**82.01**	**95.35**	**11.96**
social	**Claude 4**	11.78	2.17	9.11	19.94	73.44	63.09	89.78	19.85
	GPT-4.1	10.26	1.61	7.70	15.85	**75.78**	71.46	91.39	16.20
	Yandex	8.91	**2.04**	5.67	16.22	71.94	75.00	91.37	15.44
	Yandex+WMT	**7.83**	2.09	**4.46**	**15.76**	68.41	**79.39**	**94.59**	**15.30**
speech	**Claude 4**	17.42	3.63	12.44	30.67	69.17	50.65	88.58	30.57
	GPT-4.1	13.44	2.73	9.57	23.36	**69.77**	63.48	90.94	23.04
	Yandex	11.31	2.74	7.13	22.57	66.56	69.97	93.23	21.78
	Yandex+WMT	**10.65**	**2.66**	**6.07**	**21.11**	64.85	**73.27**	**94.60**	**19.85**

Table 11: WMT25 RATE annotation results - segment-level scores by domain.

system	errors	major	minor	MQM	accuracy	fluency	style	RATE
Claude 4	12.11	2.32	8.85	20.51	**73.20**	62.38	90.27	20.37
GPT-4.1	10.11	1.65	7.48	15.84	77.09	71.09	92.83	16.09
Yandex	**7.92**	1.85	5.08	15.01	73.11	76.64	93.12	14.45
Yandex+WMT	7.98	**1.74**	**4.65**	**14.56**	72.29	**78.90**	**94.76**	**14.15**

Table 12: WMT25 RATE annotation results - segment-level scores macro-averaged by domain.

B.2 WMT25 RATE results

system	errors	major	minor	MQM	accuracy	fluency	style	RATE
Claude 4	15.50	3.17	11.16	27.09	70.02	55.00	88.85	26.97
GPT-4.1	12.11	2.38	8.65	20.70	**71.88**	66.47	91.53	20.52
Yandex	10.13	2.46	6.36	20.04	68.58	72.29	93.07	19.23
Yandex+WMT	**9.64**	**2.35**	**5.50**	**18.87**	66.89	**75.45**	**94.73**	**17.85**

Table 13: WMT25 RATE annotation results - document-level scores.

system	errors	major	minor	MQM	accuracy	fluency	style	RATE
Claude 4	12.10	2.32	8.84	20.46	73.21	62.36	90.30	20.33
GPT-4.1	10.09	1.65	7.46	15.80	**77.16**	71.17	92.83	16.05
Yandex	**7.92**	1.86	5.08	15.02	73.08	76.61	93.10	14.46
Yandex+WMT	**7.96**	**1.74**	**4.64**	**14.52**	72.34	**78.8**4	**94.79**	**14.12**

Table 14: WMT25 RATE annotation results - document-level scores macro-averaged by domain.

severity	category	Claude 4	GPT-4.1	Yandex	Yandex+WMT
major	Consistency	0.11	0.09	0.16	0.17
	Do not translate	0.03	0.02	0.03	0.03
	Fluency	1.09	0.57	0.43	0.31
	Grammar	0.03	0.03	0.00	0.01
	Mistranslation	1.37	1.21	1.38	1.42
	NE	0.14	0.08	0.10	0.12
	Omission	0.00	0.00	0.00	0.00
	Overtranslation	0.01	0.06	0.05	0.10
	Punctuation	0.00	0.01	0.00	0.00
	Style	0.01	0.01	0.01	0.00
	Undertranslation	0.08	0.07	0.10	0.08
minor	Do not translate	0.06	0.02	0.01	0.01
	Fluency	6.19	4.85	3.59	2.80
	Grammar	1.26	0.64	0.22	0.27
	Mistranslation	1.39	1.33	1.19	1.03
	NE	0.17	0.17	0.11	0.18
	Omission	0.00	0.00	0.00	0.00
	Overtranslation	0.02	0.10	0.05	0.15
	Punctuation	0.31	0.14	0.02	0.04
	Style	0.33	0.33	0.18	0.20
	Undertranslation	0.16	0.21	0.24	0.20
trivial	Do not translate	0.00	0.00	0.00	0.00
	Fluency	0.63	0.61	0.43	0.53
	Grammar	0.02	0.03	0.02	0.05
	Mistranslation	0.06	0.05	0.04	0.07
	NE	0.00	0.01	0.04	0.02
	Omission	0.00	0.00	0.00	0.00
	Overtranslation	0.01	0.01	0.00	0.01
	Punctuation	0.01	0.01	0.00	0.00
	Style	0.02	0.03	0.02	0.01
	Undertranslation	0.01	0.00	0.02	0.01

Table 15: WMT25 RATE annotation results - segment-level error counts grouped by severity (major - 4-5, minor - 2-3, trivial - 1).