Vikhr: Constructing a State-of-the-art Bilingual Open-Source Instruction-Following Large Language Model for Russian

Aleksandr Nikolich◇
ITMO University
alexdragannikolich@gmail.com

Konstantin Korolev◇
HSE University
korolevko@icloud.com

Sergei Bratchikov
Yandex
hivaze.me@gmail.com

Igor Kiselev
Accenture
igor.kiselev@accenture.com

Artem Shelmanov
MBZUAI
artem.shelmanov@mbzuai.ac.ae

Abstract

There has been a surge in the development of various Large Language Models (LLMs). However, text generation for languages other than English often faces significant challenges, including poor generation quality and reduced computational performance due to the disproportionate representation of tokens in the model's vocabulary. In this work, we address these issues by developing a pipeline for adaptation of English-oriented pre-trained models to other languages and constructing efficient bilingual LLMs. Using this pipeline, we construct Vikhr, a state-of-the-art bilingual open-source instruction-following LLM designed specifically for the Russian language. "Vikhr" refers to the name of the Mistral LLM series and means a "strong gust of wind." Unlike previous Russian-language models that typically rely on LoRA adapters on top of English-oriented models, sacrificing performance for lower training costs, Vikhr features an adapted tokenizer vocabulary and undergoes the continued pre-training and instruction tuning of all weights. This not only enhances the model's performance but also significantly improves its computational and contextual efficiency. The remarkable performance of Vikhr across various Russian-language benchmarks can also be attributed to our efforts in expanding instruction datasets and corpora for continued pre-training. Vikhr not only sets the new state of the art among open-source LLMs for Russian but even outperforms some proprietary closed-source models on certain benchmarks. The model weights, instruction sets, and code are publicly available[1].

1 Introduction

Instruction tuning has unlocked in Large Language Models (LLMs) vast zero-shot capabilities without the need for careful prompt engineering (Ouyang et al., 2022). The most rapid research and development efforts are currently devoted to English LLMs. There has been a surge in English open-source models: Llama series (Touvron et al., 2023a,b), Mistral series (Jiang et al., 2023), Vicuna series (Chiang et al., 2023), etc. This growth is driven by the abundance of raw training data in English and dedicated efforts to create extensive sets of instruction-output pairs. Even though LLMs oriented on English have some multilingual capabilities (Zhao et al., 2024) due to the presence of small amounts of text in various languages within their training datasets (Touvron et al., 2023a), their overall performance in non-English languages remains relatively limited. Although they can usually generate portions of coherent texts, these models struggle with reasoning in non-English languages, lack culture-specific knowledge, and are highly inefficient in terms of tokenization. This inefficiency stems from how byte-pair tokenization algorithms operate, as they break down infrequent words into multiple tokens. Since multilingual data typically represents a small portion of the training dataset, non-English words are often split into many pieces. As a result, this increases the number of steps during prompt processing and text generation, reduces the effective context window, and ultimately degrades overall performance (Tikhomirov and Chernyshev, 2023; Petrov et al., 2024). This disparity places non-English languages at a disadvantage.

There is also a research direction focused on developing multilingual LLMs designed to perform well across multiple popular languages: BLOOMz (Muennighoff et al., 2023), mGPT (Shliazhko et al., 2022), Bactrian-X (Li et al., 2023), PALO (Maaz et al., 2024), Aya101 from CohereAI (Üstün et al., 2024), etc. These models are typically trained on rich multilingual datasets and are less skewed towards English. However, when aiming to perform well across multiple languages simultaneously, these models must still share their vocab-

---

◇ Equal contribution
[1]https://huggingface.co/Vikhrmodels

ulary and parameters. This often hinders their performance for each particular language in isolation, especially for the popular smaller model sizes, such as 7B and 13B.

The aim of maximizing the LLM performance for a specific language within a certain number of parameters has led researchers to develop bilingual LLMs (Sengupta et al., 2023; Pieri et al., 2024; Faysse et al., 2024). These LLMs prioritize a regional language, e.g. Jais(Sengupta et al., 2023) focuses on Arabic, but they are trained also on English data. The inclusion of English data in pre-training alongside regional language data is motivated by the significantly larger volume of English data available. This helps LLMs substantially enhance skills such as logical and common sense reasoning, which are also applied when generating text in a regional language. Bilingual LLMs is a perspective direction as they can remain small and efficient, but at the same time comprehensively capture linguistic nuances and cultural contexts of the regional language.

This work seeks to develop a pipeline for adapting English LLMs to other languages facilitating the development of bilingual LLMs. Specifically, we aim to build an instruction-following bilingual LLM for Russian and English that could be used for multilingual natural language processing research.

Russian is one of the high-resource languages and is typically represented in multilingual LLMs. Additionally, there are several proprietary closed-source LLMs, such as MTS AI, GigaChat, and YandexGPT, that meet or even surpass their English-oriented flagship competitors when it comes to text processing and generation in Russian. However, controllable research often requires white-box access to LLM logits and layer outputs, the ability to modify weights and model architecture, and consistent answers for reproducibility, which is often impossible in closed-source LLMs due to their constant development and retirement. There are only a few open-source LLMs designed for Russian: Saiga (Gusev, 2023), ruGPT (AI Forever, 2022), ruadapt (Tikhomirov and Chernyshev, 2023), and some others. Of these, only Saiga and ruadapt are instruction-tuned. We aim to fill the lack of instruction-tuned open-source LLM for Russian that is both efficient and effective.

Building even a small LLM tailored to a specific language from scratch demands a lot of computational resources. Consequently, many researchers

opt to fine-tune LoRA adapters (Hu et al., 2021) for English-oriented LLMs using some language-specific data. While this approach can improve model generation quality, it does not address computational inefficiency because the tokenizer and model vocabulary remain unchanged. In contrast, our approach not only fine-tunes a base LLM on Russian language data but also reconstructs its underlying tokenizer and vocabulary, alongside suggesting an improved method for continued pre-training. Additionally, we have significantly expanded the available Russian datasets for instruction tuning. The developed LLM achieves state-of-the-art results for the Russian language among other open-source counterparts across a wide range of benchmarks.

Contributions of the paper are the following:

- We have developed a pipeline for adapting English-oriented LLMs to other languages. The pipeline implements vocabulary adaptation, continued pre-training with regularization to prevent "catastrophic forgetting", and instruction tuning.
- Using the pipeline, we have constructed Vikhr – a state-of-the-art open-source instruction-following LLM oriented on the Russian language. In addition to its high generation quality, Vikhr features an efficient tokenizer that enables rapid text generation and good context utilization.
- We have expanded the datasets for continued pre-training of Russian language models and previously available instruction datasets.
- We have constructed two evaluation benchmarks for Russian LLMs by translating the English MMLU (Hendrycks et al., 2020) and MMLU-pro (Wang et al., 2024b) benchmarks.
- We conducted an extensive evaluation of several open-source LLMs on evaluation benchmarks for Russian, demonstrating that Vikhr achieves new state-of-the-art results. Ablation studies confirm the effectiveness and validity of the individual components within our LLM adaptation pipeline.

2 Related Work

One of the first prominent series of generative LLMs for Russian is ruGPT (AI Forever, 2022; Zmitrovich et al., 2023). The authors developed several models, trained on the standard language modeling task, with sizes reaching up to 13 billion

parameters. These models were created from the scratch and trained on a large Russian corpus, enabling them to capture the linguistic nuances of Russian more effectively than multilingual models. Additionally, since the training data was mostly in Russian, these models also have efficient tokenization. However, the lack of multilingual data (e.g. in English) limits their performance. Notably, the ruGPT models are not instruction-tuned.

Gusev (2023) suggests to leverage reasoning capabilities of existing English-oriented LLMs and adapt them to the Russian language by training LoRA adapters. They created an Alpaca-like set of Russian instruction-output pairs and performed instruction tuning on it. As a result, they have established the series of models called Saiga, which demonstrate the competitive performance and used to be a reasonable choice for an off-the-shelf open-source Russian LLMs for the past years. However, the tokenizer in these models is not adapted, so they experience issues with the context size and computational efficiency.

Tikhomirov and Chernyshev (2023) address these issues in Saiga. In addition to model tuning on Russian data, they also adapted the model tokenizer. They note that improving tokenization helps to both enhance the efficiency of the model and its performance while reducing memory consumption. However, during continued pre-training, the authors froze the model weights, except for LM heads and token embeddings, which leads to the suboptimal performance.

In this work, we take advantage of pre-trained English-oriented LLMs, adapt LLM tokenizer for better computational and contextual efficiency, leverage continued pre-training on vast Russian-language corpora with regularization for preventing "catastrophic forgetting", construct a novel extended set of Russian instruction-output pairs, and perform instruction tuning. The created LLM adaptation pipeline along with the data for continued pre-training and instruction tuning enables Vikhr to achieve new state-of-the-art results for Russian, maintain high performance for English, and demonstrate high computational efficiency.

3   LLM Construction Pipeline

The construction of Vikhr starts from one of the English-oriented LLMs. In this work, we discuss the Vikhr model based on Mistral 7B (Jiang et al., 2023). The strong logical and common-sense rea-

Content	Length	Tokenization Result
Original Sentence	31	(ru) Машинное обучение изменяет мир [(en) Machine learning changes the world.
Mistral Tokenizer	13	['Ma', 'шин', 'ное', 'об', 'у', 'чение', 'из', 'мен', 'я', 'ет', 'ми', i 'p']
Vikhr Tokenizer	7	['Ma', 'шин', 'ное', 'обуче-ние', 'изменяет', 'мир']

Table 1: A comparison of tokenization between the original Mistral model and Vikhr.
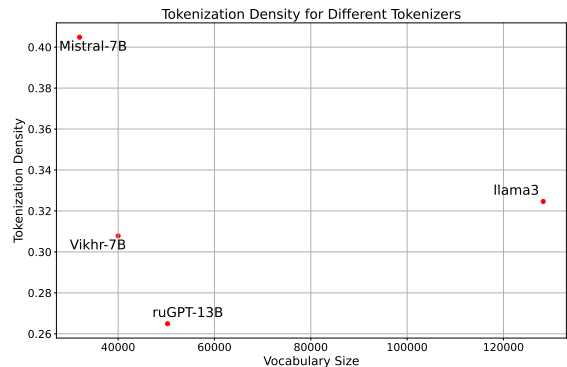

Figure 1: The efficiency of the Vikhr tokenizer for Russian in comparison to tokenizers of other models.

soning capabilities, as well as the extensive world knowledge present in Mistral LLMs provide an excellent starting point for our model. These features partially transfer to Vikhr, enhancing its performance in generating text in Russian. The process of the LLM adaptation to Russian starts with the vocabulary adaptation. Then we perform continued pre-training of the LLM on large Russian datasets to mitigate the vocabulary shift and introduce culture specific knowledge. Finally, we perform fine-tuning of Vikhr on a set of instruction-output pairs in Russian.

3.1   Vocabulary Adaptation

The big drawback of English-oriented LLMs is that each Russian word would be split into multiple tokens: a common case is when symbols in the word become individual tokens (see example in Table 1). This slows down the generation by multiple times, reduces the amount of information that could be stored in the context, and drastically hurts the generation quality.

To mitigate this problem in Vikhr, we adopt the approach suggested in (Cui et al., 2023;

Data Source	Approx. size (GB)	Tokens (Billion)
Scientific papers	20	2.5
News articles	4	1
Wikipedia	25	4
Habr	6	1
Other sources	20	2.5

Table 2: The statistics of the Russian-language datasets for continued pre-training.

Hyperparam.	Value
LR	1×10^{-3}
AdamW eps	1×10^{-8}
Num warmup steps	10
AdamW betas	0.99, 0.95
Accumulation steps	128
Batch size	3
Epochs	1
Sequence length	1024

Table 3: The hyperparameters for continued pre-training.

Tikhomirov and Chernyshev, 2023), where authors rebuild the tokenizer using a language-specific corpus. In particular, we trained a SentencePiece tokenizer (Kudo and Richardson, 2018) with a 40k vocabulary on the RuLM dataset (Gusev, 2023). As can be seen from Figure 1, the resulting tokenizer for Russian is much more efficient than the tokenizer of the original English-oriented model.

3.2 Continued Pre-training

The new vocabulary requires new embedding matrices and LM heads. The tokens that were present in the original vocabulary are initialized with the old embeddings, the new tokens are initialized by averaging the embeddings of their pieces in the original embedding matrix (Hewitt, 2021). The similar approach is also applied to LM heads. Training models with these modifications demands much more computational resources than the mainstream method of adapting LLMs to new languages using LoRA adapters (Hu et al., 2021). This is because it involves continued pre-training of the entire model and requires much more language-specific data to mitigate the shift in the vocabulary.

The dataset for continued pre-training is constructed from high-quality sources, including Russian Wikipedia, news articles, scientific papers from peer-reviewed journals and conferences, and top 100k up-voted posts on Habr – a popular online blog community focused on technology, software development, and science. The statistics of these datasets is presented in Table 2. We performed deduplication of the collection on the level of paragraphs using the MIHash algorithm (Cakir et al., 2017). Furthermore, we performed filtration of the collected data. For this purpose, we annotated 20k documents using GPT-4-turbo with the aim to assess their informativeness, usefulness for studying, grammatical correctness, style, and safety. Using these annotations, we fine-tuned a RuBERT-tiny(Dale, 2021) filtration model and applied it to

the deduplicated corpus. After filtration, the total number of tokens left for continued pre-training is 11 billion.

We observed that continued pre-training of a LLM can partially diminish the reasoning capabilities present in the original English-oriented model, significantly impacting overall performance. In our preliminary experiments, a model that underwent continued pre-training may demonstrate even worse performance on Russian benchmarks than the original English-oriented model. To alleviate this "catastrophic forgetting" in reasoning, we use the loss regularization with the KL penalty between the probability distribution of Vikhr and the reference English-oriented LLM:

$$L_{\text{Vikhr}} = L_{\text{CE}} + KL\left(P_{\text{Vikhr}} \| P_{\text{Ref}}\right). \quad (1)$$

In practice, we implement the regularization using the SLERP interpolation of model losses (Goddard et al., 2024).

To speed up the process of continued pre-training, we use an optimized Flash attention implementation[2]. As an optimization algorithm, we leverage AdamW, as it trades some memory efficiency in favor of robustness to the hyperparameter choice. The hyperparameters used for continued pre-training are presented in Table 3.

3.3 Instruction Tuning

Instruction tuning is an essential step in reaching high zero-shot performance with LLMs. It also allows to obtain a more natural communication with the model without complex prompting. Further fine-tuning techniques such as RLHF (Ouyang et al., 2022) or DPO (Rafailov et al., 2024), which require input from the assessors, are also crucial for such tasks as multicriteria alignment. However, the most significant performance gains are

---

[2]https://huggingface.co/docs/optimum/bettertransformer/tutorials/convert

still achieved through instruction tuning (Jha et al., 2023).

Previously, Gusev (2023) constructed an open-source set of instruction-output pairs for the Russian language (Saiga). The core Saiga dataset was created similar to Alpaca by querying ChatGPT (gpt-3.5-turbo) (Taori et al., 2023). In this work, we extend this set by translating two English instruction datasets. First, we translated instructions for the FLAN model (Wei et al., 2021) and generated answers in Russian using ChatGPT. Originally, FLAN instructions were constructed automatically from annotated datasets using templates to facilitate multitask and zero-shot capabilities of seq2seq models. Later, it was shown that this data also helps to improve decoder-only chat-oriented models as well. Second, we construct Veles[3] by translating the English OpenHermes (Teknium, 2023) instruction dataset. Third, we incorporate without translation Nectar[4] (Zhu et al., 2023) – an English instruction dataset. This ensures that Vikhr maintains strong performance in English as well.

Similar to the corpus for continued pre-training, we performed deduplication of the instruction set. Since the majority of the outputs were machine generated there are many low quality outputs. To mitigate this problem, we filtered out low quality pairs using a reward model trained on human data. For the reward model, we selected the multilingual-e5-large model (Wang et al., 2024a). This model was particularly suitable for our needs due to its ability to handle multilingual data efficiently, ensuring that the classifier could accurately assess the quality of responses in both Russian and English. We trained the reward model on the answer preference dataset[5], which was collected from human-written prompts and annotated using GPT-4. By applying this reward model, we filtered out low-quality instruction-output pairs, significantly enhancing the overall performance and reliability of our instruction datasets. The statistics of the Vikhr instruction datasets is presented in Table 4.

Contrary to Saiga, we do not use LoRA adapters and just as in the phase of continued pre-training, we update all model parameters. The hyperparameters for the instruction tuning phase are presented

Instruction Set	Language	# instances
Veles	Russian	30k
Nectar	English	50k
Saiga	Russian	100k
ruFLAN	Russian	500k

Table 4: The statistics of the instruction datasets.

Hyperparam.	Value
LR	1×10^{-5}
AdamW, eps	1×10^{-8}
Num warmup steps	10
AdamW, betas	0.99, 0.95
Accumulation steps	64
Batch size	3
Num epochs	3
Sequence length	1024

Table 5: The hyperparameters for instruction tuning.

in Table 5.

3.4 Hardware

Vikhr was trained on eight NVIDIA A100 GPUs 80GB. We spend approximately 1,000 GPU hours for the continued pre-training phase and 60 hours for instruction tuning.

4 Experiments

4.1 Experimental Setup

Benchmarks. The evaluation was performed on MMLU (En-MMLU) (Hendrycks et al., 2021), Ru-MMLU[6], Ru-MMLU-pro[7], ruXNLI (Conneau et al., 2018), CheGeKa (Mikhalkova and Khlyupin, 2022), Russian SuperGLUE (Shavrina et al., 2020), and MERA (Fenogenova et al., 2024). The MMLU (En-MMLU) benchmark assesses LLMs across 57 subjects through multiple-choice questions, measuring a model's general knowledge and reasoning abilities. We utilize this benchmark to verify that the model retains its bilingual proficiency. For this dataset, we report the accuracy@1 score. We constructed Ru-MMLU and Ru-MMLU-pro by automatically translating original English MMLU and MMLU-pro to Russian. Translation was done by GPT-3.5 and GPT-4 respectively. Just as for MMLU, for these datasets, we report the accuracy@1 score. CheGeKa is based on the questions from the Russian version of "Jeopardy" (the "Own

---

[3]https://huggingface.co/datasets/Vikhrmodels/Veles-2.5

[4]https://huggingface.co/datasets/berkeley-nest/Nectar

[5]https://huggingface.co/datasets/Vikhrmodels/sbs

[6]https://huggingface.co/datasets/NLPCoreTeam/mmlu_ru

[7]https://huggingface.co/datasets/Vikhrmodels/mmlupro-ru

LLM	Pre-train on Russian	Training Method	En-MMLU	Ru-MMLU	CheGeKa	Russian SuperGLUE	MERA
MTS AI Chat 7B (closed-source) ◇	false	sft+dpo	-	0.689	0.083	0.56	0.479
GigaChat-7B (closed-source) ◇	true	sft+dpo	-	0.67	0.451*	0.71*	0.479
aya101	false	pt+sft	0.41	0.37	0.005	0.36	0.320
Mistral-7B-Instruct-v0.2	false	none	0.60	0.78	0.005	0.57	0.400
rccmsu/ruadapt-mistral-7b-v0.1	false	pt+sft	0.61	0.72	0.005	0.64	0.421
rugpt13b	true	none	0.25	0.25	0.132	0.52	0.208
saiga-mistral-7b-lora	false	sft	0.60	0.76	0.223	0.64	0.442
saiga-llama3-8b	false	sft	0.59	0.78	0.225	0.66	0.476
(Ours) Vikhr-7B-instruct_0.2	true	pt+sft	**0.62**	**0.80**	**0.231**	**0.67**	**0.485**

Table 6: Evaluation results for Russian and multilingual LLMs. Pre-train on Russian means that the model underwent (continued) pre-training on Russian data. The following abbreviations are used: sft – instruction tuning, pt – (continued) pre-training; dpo – direct preference optimization. ◇ The results for GigaChat and MTS AI are taken from the leaderboards. The best result among open-source models is highlighted with bold, the second best is underscored. The best result among closed-source proprietary models is marked with *.

Prompt	Answer of base Mistral-0.2-Instruct	Answer of Vikhr
Что такое чебурек? [What does cheburek means?]	Чебурек - это классическое dish среднеазиатской и русской кухни, которое включает в себя тонкий lavash, завернутый в форме полумесяца или кольца. Он обычно наполнен meat (говядиной, бараниной или курицей) и cheese. История чебурека начинается с древних тюркских племен, которые готовили подобные dishes из мяса и теста. ... Чебурек часто подается как горячее dish, жареное на сковороде или в духовке. Тесто обычно тонкое и хрустящее снаружи, а внутри мягкое и сочное. Мясо и сыр добавляют flavor depth и питательную ценность.	Чебурек — это традиционное блюдо кавказской кухни, которое представляет собой большой, тонкий пирог из пресного теста с мясной начинкой, обжаренный во фритюре. Внутри чебурека обычно находится сочный фарш из баранины или говядины, приправленный луком, солью, перцем и другими специями. Благодаря обжарке в масле, тесто становится хрустящим снаружи, а начинка остается мягкой и сочной. История чебурека уходит корнями в древние времена, когда кочевники готовили его на раскалённых камнях или в больших котлах с горячим маслом.
Расскажи на русском языке что такое хачапури [Tell in Russian what hatchapuri is]	Хачапури (хачапури, хачъапури) – это традиционное блюдо грузинской кухни. Это тип пицце или за Francia, выполненной в виде круглой или полукруглой лепёшки, покрытой верху layerm layerim – смесью тонко нарезанныхimo эplesses, лука ..	Хачапури — это традиционное грузинское блюдо, представляющее собой лепёшку с сыром внутри. Вот рецепт хачапури. Ингредиенты: - Для теста: - 3 стакана муки - 1 ч. л. соли - 1 ч. л. сахара - 100 мл теплой воды - Для смазывания: 1 ст. л. топленого масла ...
Расскажи на русском языке как приготовить уху [Tell in Russian how to cook a fish soup]	Уха (из Рыбы) – традиционное блюдо русской кухни, которое готовится predominantly из щуки или окуня. Рас**те процедуру приготовления ухи в домашних условиях: Выберите чистую рыбу. Очистить её от шерсти иICScone всех мелких костей s...	Уха — это традиционное русское блюдо, представляющее собой наваристый суп из рыбы, который варится на медленном огне. Вот базовый рецепт ухи, который можно адаптировать под свои вкусовые предпочтения. Ингредиенты: 1 кг свежей рыбы (можно использовать несколько видов, например, карп, щука, судак) 3-4 картофелины 1 крупная луковица 2-3 моркови ...

Table 7: Comparison of texts generated by the base LLM Mistral-0.2-Instruct (on the left) with texts generated by our model Vikhr (on the right). Issues in the output are highlighted with red.

game"). This benchmark contains challenging open-ended questions, requiring logical reasoning and world knowledge. It includes 29,376 training and 416 test instances. The reported evaluation metric is the F1 score. Russian SuperGLUE is a benchmark similar to well-known English SuperGLUE (Wang et al., 2019). It tests LLMs on various natural language understanding tasks like reading comprehension and textual entailment. The metric reported in the results is accuracy@1. The MERA benchmark encompasses 21 evaluation tasks for generative LLMs in 11 skill domains. Note that among other tasks, MERA also includes CheGeKa, RuMMLU, and one of the subtasks of SuperGLUE (RWSD). The reported evaluation metric is the to-tal score, which is the average of scores across all non-diagnostic tasks.

Baselines. We compare Vikhr to six open-source and two proprietary closed-source competitors of the similar size. Open-source models: aya101 (Üstün et al., 2024) – a massively multilingual LLM from CohereAI that follows instructions in 101 languages[8], it shows state-of-the-art results among massively multilingual LLMs; Mistral-7B-0.2-instruct (Jiang et al., 2023) – an English-oriented LLM that was used as the base model for Vikhr; rccmsu/ruadapt_mistral_saiga_7b_v0.1 (Tikhomirov and Chernyshev, 2023) – a Russian-

[8]https://huggingface.co/CohereForAI/aya-101

oriented LLM that was constructed from the Mistral model using similar adaptations of the tokenizer, token embeddings, and the LM head; saiga-mistral-7b-lora and saiga-llama3-8b (Gusev, 2023) – two versions of the Saiga models based on English-oriented LLMs and obtained by fine-tuning LoRA adapters on the Saiga instruction dataset[9]. Closed-source proprietary models for Russian: MTS AI Chat[10] and GigaChat-7b. The access to GigaChat weights is closed, so the reported results are taken from the leaderboards[11]. The results of MTS AI Chat are also taken from the leaderboard[12].

4.2 Results in NLU Tasks

The evaluation results are presented in Table 6. As we can see, Vikhr outperforms all open-source models, including the ones that were built specifically for Russian. It also slightly outperforms its parent model Mistral on the En-MMLU benchmark, which might be the result of longer pre-training. The second place with close scores for all 4 Russian language benchmarks is obtained by the Saiga model based on recently released Llama-3. The high scores of this model probably are the result of the transfer of the outstanding performance of Llama-3. Since Saiga based on Llama-3 outperforms Saiga based on Mistral, we expect that applying our adaptation pipeline to Llama-3 would also help further improving the state of the art.

We note that the original Mistral-7B-0.2-instruct, despite being an English-oriented model, demonstrates competitive performance in 3 out of 4 Russian benchmarks. This demonstrates that such models could be viable alternatives at least for NLU tasks. The only dataset, where its performance is very low is CheGeKa, which is related to open-ended question-answering. This may be due to the lack of culture-specific knowledge, as the English-oriented model has not seen much Russian texts. Note that the MTS AI Chat also shows very low results on CheGeKa, which might also indicate the lack of culture-specific knowledge.

The proprietary model GigaChat substantially outperforms Vikhr on CheGeKa and notably on Russian SuperGLUE. We assume this is due to the use of much larger Russian datasets for pre-training. However, surprisingly, it falls behind Vikhr on Ru-

Vocab. Size	PPL↓	ruXNLI↑
33k	**10.2**	0.42
<u>40k</u>	14.4	**0.46**
60k	16.7	0.43
80k	20.4	0.41

Table 8: Performance of the intermediate Vikhr models with different vocabulary sizes after continued pre-training. The base model is LLaMa-2 7b. The vocabulary size selected for the final Vikhr model is underlined.

	PPL↓	ruXNLI↑	Ru-MMLU-pro↑
No filt. (17b tokens)	8.4	0.37	10.2
With filt. (11b tokens) (Ours)	**7.2**	**0.46**	**11.1**

Table 9: Performance metrics with and without filtration of the corpus for the continued pre-training. Perplexity is computed on the instruction dataset.

MMLU. On all benchmarks, Vikhr outperforms the proprietary competitor from MTS AI.

4.3 Quality of Generated Text

Previous results demonstrate the performance of our model in NLU tasks. However, the performance in NLU does not always reflect the performance in text generation, as the former is based on classification and ranking capabilities of models.

To check the performance of the model in text generation, we performed qualitative analysis of LLM outputs. Table 7 compares several responses of Vikhr with outputs of the base model (Mistral-7B-Instruct-v0.2). As we can see from the presented examples, when Mistral generates Russian text, it often injects English words. Moreover, sometimes generated words consist of an English and a Russian token. From the second example, we see that Mistral also has issues with grammatical coherence. In the third example, LLM suggests to "clean a fish from fur", which illustrates the lack of understanding of word meanings in the Russian language. We also note that Mistral tends to answer in English even when the input prompt is in Russian. These issues appear very often and make the base model useless for generation of Russian texts in practical scenarios. On the considered examples, Vikhr does not demonstrate any of these problems. Texts generated by Vikhr are grammatically coherent and correct.

4.4 Ablation Studies

We conducted several ablation studies to demonstrate effects of various features of our model translation pipeline that allowed Vikhr to achieve high

---

	PPL↓	ruXNLI↑
No regul.	8.1	0.34
With regul. (Ours)	**7.2**	**0.45**

Table 10: Performance with and without KL loss regularization during continued pre-training.

Instruction Set	Ru-MMLU-pro
Saiga SFT	0.21
Translated Nectar	0.20
Ours	**0.27**

Table 11: Performance of Vikhr models fine-tuned on various instruction sets. The base model is Llama-3 8b.

performance.

Selection of the vocabulary size. Table 8 presents the performance of intermediate Vikhr models with different vocabulary sizes after the stage of continued pre-training. We measure perplexity of the LLM on the instruction set and the performance on the ruXNLI task (Conneau et al., 2018). As we can see, perplexity increases with the vocabulary size, indicating some degradation. However, the performance on the ruXNLI dataset is not monotonic. While the results for the largest vocabulary size are lower than those for the smallest, we observe a performance peak at a vocabulary size of 40k tokens. We selected this size as it offers improvements in the final task with only a slight increase in perplexity.

Effect of filtration of the corpus for continued pre-training is illustrated in Table 9. As we can see, despite reducing the size of the data, performing continued pre-training on the filtered corpus results in a model with lower perplexity and substantially better scores in both considered end tasks: Ru-MMLU-pro and ruXNLI. This again highlights the importance of data quality for constructing good LLMs.

Effect of loss regularization in continued pre-training is illustrated in Table 10. The results show that the KL regularization introduced in our pipeline slightly reduces perplexity and substantially increases the model performance in the ruXNLI task. This shows that continued pre-training on its own might deteriorate LLM reasoning capabilities, and proper regularization helps preventing the catastrophic forgetting.

Effect of fine-tuning on various instruction sets is illustrated in Table 11. The results demonstrate that fine-tuning on our instruction set gives a big boost in performance on NLU tasks compared to Saiga and translated Nectar.

5 Conclusion

We have presented Vikhr – a new state-of-the-art open-source instruction-following bilingual LLM oriented on the Russian language. To create Vikhr, we developed a comprehensive pipeline for adapting English-oriented LLMs to other languages. The pipeline includes the adaptation of the tokenizer vocabulary, continued pre-training of the entire model, and instruction tuning. We have also constructed a new dataset for instruction tuning by expanding the Saiga dataset with automatically translated and cleaned English instruction datasets. Our extensive work enabled Vikhr to outperform the known baselines while maintaining computational efficiency.

We hope that the developed cross-lingual adaption pipeline and the published models will foster the research on LLMs and enhance the diversity of languages incorporated into research agendas.

In the future work, we plan to release in the open-source new better versions of Vikhr. At the moment, our best publicly available model is Vikhr-Nemo-12B-Instruct[13] based on Mistral NeMo. We also plan to perform cross-lingual adaption of LLMs to low-resource languages such as Belarusian, Serbian, and Kazakh.

Limitations

We do not introduce additional restrictions to the usage of our models. However, the users must comply with the license of the base model and instruction datasets.

We do not implement RLHF / DPO fine-tuning of Vikhr due to the lack of the resources for human annotation. We expect further performance improvements from these techniques.

We do not introduce additional instruction-output pairs to facilitate LLM alignment. However, we note that the majority of the data for supervised fine-tuning of Vikhr are obtained from the Chat-GPT model series, so our model partially inherits its alignment.

---

[13]https://huggingface.co/Vikhrmodels/
Vikhr-Nemo-12B-Instruct-R-21-09-24

Ethical Considerations

The development and deployment of Vikhr raise several ethical considerations that must be addressed to ensure its responsible use:

- Bias and Fairness: For developing Vikhr, we use publicly available data. Despite efforts to train Vikhr on diverse datasets, there is a risk of inherent biases in the data which may be reflected in the model's outputs. Continued monitoring and evaluation are required to mitigate any biases, ensuring fair and unbiased performance.

- Misinformation: As with any LLM, Vikhr has the potential to generate misleading or incorrect information. It is crucial to establish guidelines and mechanisms for users to verify the information provided by the model, promoting critical assessment and cross-referencing with reliable sources.

- Misuse: Vikhr can be used for malicious purposes, such as generating harmful content, spam, or deepfakes. Implementing usage restrictions and monitoring mechanisms to detect and prevent misuse is critical to safeguard against these risks.

References

AI Forever. 2022. ru-gpts: Generative pre-trained transformer models for russian. https://github.com/ai-forever/ru-gpts.

Fatih Cakir, Kun He, Sarah A. Bargal, and Stan Sclaroff. 2017. Mihash: Online hashing with mutual information. In *IEEE International Conference on Computer Vision (ICCV)*.

Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E. Gonzalez, Ion Stoica, and Eric P. Xing. 2023. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality.

Alexis Conneau, Ruty Rinott, Guillaume Lample, Adina Williams, Samuel R. Bowman, Holger Schwenk, and Veselin Stoyanov. 2018. Xnli: Evaluating cross-lingual sentence representations. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Yiming Cui, Ziqing Yang, and Xin Yao. 2023. Efficient and effective text encoding for chinese llama and alpaca. *arXiv preprint arXiv:2304.08177*.

David Dale. 2021. Small and fast bert for russian. [Online; posted 10-June-2021].

Manuel Faysse, Patrick Fernandes, Nuno Guerreiro, António Loison, Duarte Alves, Caio Corro, Nicolas Boizard, João Alves, Ricardo Rei, Pedro Martins, et al. 2024. Croissantllm: A truly bilingual french-english language model. *arXiv preprint arXiv:2402.00786*.

Alena Fenogenova, Artem Chervyakov, Nikita Martynov, Anastasia Kozlova, Maria Tikhonova, Albina Akhmetgareeva, Anton Emelyanov, Denis Shevelev, Pavel Lebedev, Leonid Sinev, Ulyana Isaeva, Katerina Kolomeytseva, Daniil Moskovskiy, Elizaveta Goncharova, Nikita Savushkin, Polina Mikhailova, Anastasia Minaeva, Denis Dimitrov, Alexander Panchenko, and Sergey Markov. 2024. MERA: A comprehensive LLM evaluation in Russian. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9920–9948, Bangkok, Thailand. Association for Computational Linguistics.

Charles Goddard, Shamane Siriwardhana, Malikeh Ehghaghi, Luke Meyers, Vlad Karpukhin, Brian Benedict, Mark McQuade, and Jacob Solawetz. 2024. Arcee's mergekit: A toolkit for merging large language models. *arXiv preprint arXiv:2403.13257*.

Ilya Gusev. 2023. rulm: A toolkit for training neural language models. https://github.com/IlyaGusev/rulm.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2020. Measuring massive multitask language understanding. In *International Conference on Learning Representations (ICLR)*.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2021. Measuring massive multitask language understanding. *Proceedings of the International Conference on Learning Representations (ICLR)*.

John Hewitt. 2021. Initializing new word embeddings for pretrained language models.

Edward J Hu, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. 2021. Lora: Low-rank adaptation of large language models. In *International Conference on Learning Representations*.

Aditi Jha, Sam Havens, Jeremy Dohmann, Alex Trott, and Jacob Portes. 2023. Limit: Less is more for instruction tuning across evaluation paradigms. *arXiv preprint arXiv:2311.13133*.

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. 2023. Mistral 7b. *arXiv preprint arXiv:2310.06825*.

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.

Haonan Li, Fajri Koto, Minghao Wu, Alham Fikri Aji, and Timothy Baldwin. 2023. Bactrian-x: A multilingual replicable instruction-following model with low-rank adaptation. *arXiv preprint arXiv:2305.15011*.

Muhammad Maaz, Hanoona Rasheed, Abdelrahman Shaker, Salman Khan, Hisham Cholakal, Rao M Anwer, Tim Baldwin, Michael Felsberg, and Fahad S Khan. 2024. Palo: A polyglot large multimodal model for 5b people. *arXiv preprint arXiv:2402.14818*.

Elena Mikhalkova and Alexander A. Khlyupin. 2022. Russian jeopardy! data set for question-answering systems. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 508–514, Marseille, France. European Language Resources Association.

Niklas Muennighoff, Thomas Wang, Lintang Sutawika, Adam Roberts, Stella Biderman, Teven Le Scao, M Saiful Bari, Sheng Shen, Zheng Xin Yong, Hailey Schoelkopf, et al. 2023. Crosslingual generalization through multitask finetuning. In *The 61st Annual Meeting Of The Association For Computational Linguistics*.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744.

Aleksandar Petrov, Emanuele La Malfa, Philip Torr, and Adel Bibi. 2024. Language model tokenizers introduce unfairness between languages. *Advances in Neural Information Processing Systems*, 36.

Sara Pieri, Sahal Shaji Mullappilly, Fahad Shahbaz Khan, Rao Muhammad Anwer, Salman Khan, Timothy Baldwin, and Hisham Cholakkal. 2024. Bimedix: Bilingual medical mixture of experts llm. *arXiv preprint arXiv:2402.13253*.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. 2024. Direct preference optimization: Your language

model is secretly a reward model. *Advances in Neural Information Processing Systems*, 36.

Neha Sengupta, Sunil Kumar Sahu, Bokang Jia, Satheesh Katipomu, Haonan Li, Fajri Koto, Osama Mohammed Afzal, Samta Kamboj, Onkar Pandit, Rahul Pal, et al. 2023. Jais and jais-chat: Arabic-centric foundation and instruction-tuned open generative large language models. *arXiv preprint arXiv:2308.16149*.

Tatiana Shavrina, Alena Fenogenova, Emelyanov Anton, Denis Shevelev, Ekaterina Artemova, Valentin Malykh, Vladislav Mikhailov, Maria Tikhonova, Andrey Chertok, and Andrey Evlampiev. 2020. Russiansuperglue: A russian language understanding evaluation benchmark. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4717–4726.

Oleh Shliazhko, Alena Fenogenova, Maria Tikhonova, Vladislav Mikhailov, Anastasia Kozlova, and Tatiana Shavrina. 2022. mgpt: Few-shot learners go multilingual. *arXiv preprint arXiv:2204.07580*.

Rohan Taori, Ishaan Shum, Pieter Abbeel, Carlos Guestrin, and Percy Liang. 2023. Stanford alpaca: An instruction-following language model. *GitHub*.

Teknium. 2023. Openhermes 2.5: An open dataset of synthetic data for generalist llm assistants.

Mikhail Tikhomirov and Daniil Chernyshev. 2023. Impact of tokenization on llama russian adaptation. *arXiv preprint arXiv:2312.02598*.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. 2023a. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023b. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Ahmet Üstün, Viraat Aryabumi, Zheng Yong, Wei-Yin Ko, Daniel D'souza, Gbemileke Onilude, Neel Bhandari, Shivalika Singh, Hui-Lee Ooi, Amr Kayid, Freddie Vargus, Phil Blunsom, Shayne Longpre, Niklas Muennighoff, Marzieh Fadaee, Julia Kreutzer, and Sara Hooker. 2024. Aya model: An instruction finetuned open-access multilingual language model. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15894–15939, Bangkok, Thailand. Association for Computational Linguistics.

Alex Wang, Yada Pruksachatkun, Nikita Nangia, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy,

and Samuel Bowman. 2019. Superglue: A stickier benchmark for general-purpose language understanding systems. *Advances in neural information processing systems*, 32.

Liang Wang, Nan Yang, Xiaolong Huang, Linjun Yang, Rangan Majumder, and Furu Wei. 2024a. Multilingual e5 text embeddings: A technical report. *arXiv preprint arXiv:2402.05672*.

Yubo Wang, Xueguang Ma, Ge Zhang, Yuansheng Ni, Abhranil Chandra, Shiguang Guo, Weiming Ren, Aaran Arulraj, Xuan He, Ziyan Jiang, et al. 2024b. MMLU-pro: A more robust and challenging multi-task language understanding benchmark. *arXiv preprint arXiv:2406.01574*.

Jason Wei, Maarten Bosma, Vincent Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M Dai, and Quoc V Le. 2021. Finetuned language models are zero-shot learners. In *International Conference on Learning Representations*.

Jun Zhao, Zhihao Zhang, Qi Zhang, Tao Gui, and Xuanjing Huang. 2024. Llama beyond english: An empirical study on language capability transfer. *arXiv preprint arXiv:2401.01055*.

Banghua Zhu, Evan Frick, Tianhao Wu, Hanlin Zhu, and Jiantao Jiao. 2023. Starling-7b: Improving llm helpfulness & harmlessness with rlaif.

Dmitry Zmitrovich, Alexander Abramov, Andrey Kalmykov, Maria Tikhonova, Ekaterina Taktasheva, Danil Astafurov, Mark Baushenko, Artem Snegirev, Tatiana Shavrina, Sergey Markov, et al. 2023. A family of pretrained transformer language models for russian. *arXiv preprint arXiv:2309.10931*.