Записки научных
семинаров ПОМИ
Том 546, 2025 г.

A. Latushko, E. Bruches

RUMATHBERT: A RUSSIAN-LANGUAGE MODEL FOR MATHEMATICAL FORMULA INTERPRETATION

Abstract. Important information in scientific and technical texts is often contained within mathematical formulae and cannot be acquired from the plain text, making it a significant challenge for vanilla language models to process such texts containing formulae in a manner that fully encapsulates their semantics. While there have been models developed for this purpose for the English language, none have been created for Russian so far. In this paper we present RuMathBERT: a model trained on Russian texts, which can be used for processing scientific texts containing formulae. Evaluating our model quality and comparing it to other models used for processing Russian and English regular and scientific texts demonstrated that RuMathBERT shows better understanding of the semantics of formulae and their relationship with context. The dataset used for training the model is available on Hugging Face at `https://huggingface.co/datasets/iis-research-team/ruwiki-formulae` and the RuMathBERT model itself is also available at `https://huggingface.co/iis-research-team/RuMathBERT`.

§1. Introduction

Mathematical formulae play a significant role across various disciplines such as science, technology, and engineering. Research efforts focusing on mathematical formulae for different tasks such as Mathematical Information Retrieval (MIR) [16] and Mathematical Formula Understanding (MFU) [3, 10], have consistently drawn considerable interest from researchers. Despite this, processing mathematical information remains a complex challenge, largely due to the wide variety of ways mathematical formulae can be represented, their intricate structures, and the ambiguities associated with their underlying meanings.

For scientific texts formulae play a crucial role as they contain a lot of information which cannot be acquired from the plain text. They cannot be ignored during the text processing, removing them may make the text unreadable and meaningless. But on the other side, vanilla language models

Key words and phrases: BERT, mathematical texts, formulae, NLP.

were trained on the raw texts and almost did not include formulae in the training set. However, their presence is extremely important and such special models are needed to be developed.

There are several ways of representing mathematical formulae in the texts: Symbol Layout Tree (SLT), Operator Tree (OPT), LaTeX. SLTs represent formula appearance by the spatial arrangements of math symbols, while OPTs define the mathematical operations represented in expressions. LaTeX has special syntax for the mathematical expressions. In this work we use LaTeX representation for the formulae as it is an easy way of incorporating this type of information into natural language texts.

To pre-train the scientific model, we created a dataset which contains texts in the Russian language with the incorporated formulae in LaTeX encoding. The dataset was collected from the Wikipedia texts and covers the Math domain. Each sample contains the mathematical formula in LaTeX encoding and its context in natural language. We propose that these two types of information help to better learn the underlying semantics.

Also, we pre-trained the BERT model on this dataset and showed that this model better copes with the tasks which require knowledge from the mathematical formulae.

The main contributions of this work are summarized as follows:

- The dataset to pre-train language model which contains samples with mathematical formulae in LaTeX and their surrounding context in Russian language.
- The pre-trained RuMathBERT model which is open-sourced and may be used by other researchers.
- The dataset with mapping of mathematical formulae on their trivial names, which can be used to evaluate the models quality.

§2. Related Work

Pre-trained language models such as ELMo [14], BERT [5] etc. have revolutionized the NLP. Such models are the strong base for solving different tasks in natural language, e.g. text classification, named entity recognition, machine translation etc.

While being extremely useful in NLP, these models were adapted to solve tasks which include not only texts in natural language but also coding languages (e.g. CodeBERT [6]), chemical formulae (e.g. Chem-BERTa [1], SMILES-BERT [15], MolBERT [7]) or mathematical formulae (e.g. MathBERT [13]). Some of them propose new training objective tasks to better acquire new types of information.

Scientific texts cover many domains, including math, information technology, biology, chemistry and many others. And all these fields have specific token inclusions. This led to the development of foundation models for science texts. For example, SciBERT [2] was pre-trained on scientific texts and showed better performance on specific tasks compared to vanilla BERT.

While most models are developed for the English language, adapting and developing models for low-resource languages is still an important direction for this area. For the Russian language RuSciBERT [8] was developed to process scientific texts. This model is general for many domains, and pre-trained models to process mathematical texts with an abundance of formulae for Russian language are still of interest.

We propose RuMathBERT, which is a pre-trained model for mathematical texts in the Russian language. It is capable of processing code-mixed texts, which contain both text and LaTeX tokens.

§3. Data Collection

This section outlines the process of dataset creation and the corresponding criteria for selecting data samples and processing them according to the project's requirements. At this stage of the project, our primary objective is to collect a sufficient number of Russian text samples containing mathematical formulae for their use in pre-training BERT.

In this study, a formula is defined as a mathematical expression containing at least one variable and at least one operation.

3.1. Formula processing.

3.1. Formula processing. Since the central focus of our approach is the semantic and syntactic correspondence between plain text written in a natural language and mathematical formulae, we must consider the following aspects.

Formula representation: to maintain consistency across samples, it is necessary for the formulae to be represented in a single format throughout the dataset, which should be capable of reflecting both the syntax and the semantics of the formula. As stated earlier, several ways of formula representation, such as LaTeX, OPTs and SLTs, meet these requirements, with LaTeX being a particularly suitable choice for this study. Its advantages include its widespread use across all scientific fields and its ability to unambiguously encode a formula's structure.

Relationship with context: to capture the relationship between the formal representation of formulae and natural language descriptions, each sample must include the immediate context in which the formula appears inside a document. To comply with BERT requirements for maximum input length of 512 tokens while preserving sufficient context, we limit the left context to the interval between the beginning of the paragraph containing a formula up to the formula itself, whereas the right context is limited to at most two sentence breaks after the formula.

3.2. Collecting data.

The dataset is sourced from the Russian segment of Wikipedia, specifically the articles under the Math domain, mainly in the following subcategories:

(1) Algebra and calculus;
(2) Mathematical logic and set theory;
(3) Probability theory and mathematical statistics;
(4) Matrix theory.

In the HTML structure of Wikipedia articles, mathematical expressions are enclosed within a <span> element with the class “mwe-math-element” and can be extracted in LaTeX format from the alttext attribute of a <math> element, allowing to locate both the formula and its context. We processed a total of 19 869 articles, extracting 191 173 samples of formulae in context from both inline and block display modes.

3.3. Post-processing and dataset preparation.

For dataset integrity we used the display math mode delimiters to format the main formula and the inline math mode delimiters to format any additional mathematical expressions found in the context. This formatting ensures that extracted text samples can be directly rendered as a LaTeX document. An example of a processed sample is presented in Table 1.

After removing the samples that do not fit the requirements of having at least one operation and at least one variable, as well as removing duplicate samples, we are left with a dataset containing a total of 190 898 samples with 117 582 unique formulae.

Table 1. Example sample from the testing dataset.

left_context	**formula**	**right_context**
Статистика критерия согласия `\(\chi ^{2}\)` Пирсона определяется соотношением	`\chi ^{2}=n\sum _{i=1}^{k}{\frac {\left(n_{i}/n-P_{i}(\theta )\right)^{2}}{P_{i}(\theta )}}`	. В случае проверки простой гипотезы, в пределе при `\(n\to \infty \)` эта статистика подчиняется `\(\chi _{r}^{2}\)`-распределению с `\(r=k-1\)` степенями свободы, если верна проверяемая гипотеза `\(H_{0}\)`.

§4. Experiments

In this section we discuss the experiments conducted on the dataset and the methods of evaluating the quality of the RuMathBERT model pre-trained on the dataset.

4.1. Model training. We configure the BERT model with a maximum sequence length of 512 tokens and a custom WordPiece tokenizer trained on the same dataset with a vocabulary size of 30 000 and additional special tokens [BOF] and [EOF] for marking the beginning and the end of formulae.

We pre-train the RuMathBERT model using a batch size of 8 for 5 epochs, applying both MLM (15% masking probability) and NSP objectives.

4.2. Testing dataset. For measuring the model performance we collect a dataset for testing, consisting of 100 formulae, with the majority of them coming from the same fields of mathematics as the main dataset. Each formula in the testing dataset is matched with its name that is commonly used in its area of application, as well as its topics (the sub-domains of mathematics it belongs to), as seen in Table 2.

4.3. Metrics. We implement the following methods of assessing RuMathBERT quality.

Table 2. Example samples from the testing dataset.

formula	name	topic
`P(A\mid B)={\frac {P(B\mid A)\,P(A)}{P(B)}}`	Формула Байеса	Теория вероятностей, Байесовская статистика
`P=a\oplus \bigoplus_{\begin{array}{c}1\leqslant i_{1}<\ldots<i_{k}\leqslant n\\k\in{\overline{1,n}}\end{array}}a_{i_{1},\ldots,i_{k}}\wedgex_{i_{1}}\wedge \ldots\wedgex_{i_{k}},\quad a,a_{i_{1},\ldots ,i_{k}}\in \{0,1\}`	Полином Жегалкина	Булева алгебра, Математическая логика, Теория дискретных функций

Cosine similarity: first, we calculate the cosine similarity (S_C) between the formula vectors (A_f) and formula (as well as topic and context) name vectors (B_n) from the testing dataset. The following metrics are described for one representative case (formula names) out of the three ones, as they are analogous for topics and contexts.

Using cosine similarity, we calculate (1) the mean similarity between the formulae and their correct names; (2) the mean similarity between all formulae and all names and (3) the difference between them, which reflects how well the model understands the semantic relationship between correct formula-name pairs compared to all possible formula-name pairs.

(1)

$$\overline{S_{\text{correct}}(A_f, B_n)} = \frac{1}{100} \sum_{\substack{f=1 \\ n=f}}^{100} S_C(A_f, B_n)$$

(2)

$$\overline{S_{\text{all}}(A_f, B_n)} = \frac{1}{10000} \sum_{\substack{f=1 \\ n=1}}^{100} S_C(A_f, B_n)$$

(3)

$$\Delta_{\text{correct - all}} = \overline{S_{\text{correct}}(A_f, B_n)} - \overline{S_{\text{all}}(A_f, B_n)}$$

Average rank of the correct name: (4) based on sorting the name vectors according to their similarity to the formula vector we locate the position of the correct name vector (1 is the most similar, while 100 is the least similar) and calculate the mean value across all 100 formulas.

(4)

$$\overline{R_{correct}(A_f, B_n)} = \frac{1}{100} \sum_{\substack{f=1 \\ n=f}}^{100} \text{rank}\,(S_C(A_f, B_n))$$

acc@5: (5) we calculate the ratio of correct names in the top 5 closest name vectors for each formula.

(5)

$$acc@5 = \sum_{\substack{f=1 \\ n=f}}^{100} 1\,(n = f \wedge rank(S_C(A_f, B_n)) \leqslant 5)$$

We compared our RuMathBERT model with four other BERT-based models: tbs17/MathBERT-custom [13], bert-base-cased [5], DeepPavlov /rubert-base-cased [11] and ai-forever/ruSciBERT [8] (all other models weights are loaded from HuggingFace). We also fine-tuned the MathBERT-custom and the BERT multilingual base model (uncased) models. The resulting metrics for the formulae names can be seen in Table 3).

Table 3. Performance results for formulae names.

Model	**acc@5** ↑	$\boldsymbol{\Delta_{\textbf{correct}-\textbf{all}}}$ ↑	**Average rank** ↓
DeepSeek	**0.99**	**0.7717**	**1.44**
GPT-4o-mini	0.89	0.7498	3.92
GigaChat	0.74	0.7173	7.02
RuMathBERT	0.22	0.0492	31.95
ft_MathBERT	0.13	0.0362	36.04
ft_bert-base-mltl	0.1	0.0218	44.7
bert-base-cased	0.06	0.0002	50.5
ruSciBERT	0.06	0.0019	49.12
rubert-base-cased	0.04	0.0014	50.02
MathBERT	0.03	0.0005	49.78

To assess the maximum achievable quality on this task, we also evaluated several state-of-the-art LLMs: DeepSeek [4], GigaChat [9], and GPT-4o-mini [12]. Notably, these models likely achieve high metrics due to encountering similar examples during pretraining. In contrast, BERT-based models are less likely to be trained on such structural elements from Wikipedia articles, which may explain their lower performance. LLMs' results establish a target for RuMathBERT's further optimization.

It can be observed that Δ displays a clear difference in orders of magnitude among the BERT-based models: those not specifically pre-trained on Russian texts (MathBERT-custom and bert-base-cased, 10^{-4}), those pre-trained on Russian texts (rubert-base-cased and ruSciBERT, 10^{-3}), as well as RuMathBERT and the fine-tuned models (10^{-2}), with RuMathBERT having the evident advantage over the other models. The largest margin between correct-pair and overall-pair similarity shows that RuMathBERT possesses a better understanding of formulae semantics.

Regarding the average rank of the correct name metric, there is no strong evidence that the BERT-based models other than RuMathBERT produce meaningful results, as their average ranks are between 49.12 and 50.5, close to those expected from a random distribution of answers.

§5. Conclusion

In this study, we present RuMathBERT, a BERT-based model pre-trained on Russian texts containing mathematical formulae in LaTeX encoding. The evaluation of the model's quality showcases that while our model's mean cosine similarity values are not as high as some of the other BERT-based models, there is the largest difference of 0.0492 between the mean similarity score of correct formula-name pairs and the mean similarity score of all formula-name pairs, which shows a better understanding of the semantic relationship between LaTeX-formatted formulae and and their context. RuMathBERT also demonstrates a clear advantage in the average rank of the correct name, which is 31.95, while the other BERT-based models have this value around the average value expected from random guessing.

One limitation of our approach is sourcing the entire dataset from Wikipedia's Math domain only, which limits the model's applicability to other domains and subfields of mathematics, which are not represented in this selection. Another limitation is the use of the LaTeX format, which, although a great representation of formulae semantics, is limited in its capability of representing the hierarchical structure of formulae.

Future plans include developing a data augmentation strategy based on replacing sub-formulae with equivalent expressions for further semantic variability, as well as developing a tool for transforming LATEX formatted formulae into OPTs.

References

1. W. Ahmad, E. Simon, S. Chithrananda, G. Grand, and B. Ramsundar, *ChemBERTa-2: Towards Chemical Foundation Models*, ArXiv preprint arXiv:2209.01712 (2022). Available: https://arxiv.org/abs/2209.01712.
2. I. Beltagy, K. Lo, and A. Cohan, *SciBERT: A Pretrained Language Model for Scientific Text*, ArXiv preprint arXiv:1903.10676 (2019). Available: https://arxiv.org/abs/1903.10676.
3. K. Davila and R. Zanibbi, *Layout and Semantics: Combining Representations for Mathematical Formula Search*, in: Proceedings of the ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR), 2017, pp. 1165–1168.
4. DeepSeek-AI, *DeepSeek-V3 Technical Report*, ArXiv preprint arXiv:2412.19437 (2025). Available: https://arxiv.org/abs/2412.19437.
5. J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, *BERT: Pre-Training of Deep Bidirectional Transformers for Language Understanding*, ArXiv preprint arXiv:1810.04805 (2018).
6. Z. Feng, D. Guo, D. Tang, N. Duan, X. Feng, M. Gong, L. Shou, B. Qin, T. Liu, D. Jiang, and M. Zhou, *CodeBERT: A Pre-Trained Model for Programming and Natural Languages*, ArXiv preprint arXiv:2002.08155 (2020). Available: https://arxiv.org/abs/2002.08155.
7. B. Fabian, T. Edlich, H?l?na Gaspar, M. Segler, J. Meyers, M. Fiscato, and M. Ahmed, *Molecular Representation Learning with Language Models and Domain-Relevant Auxiliary Tasks*, ArXiv preprint arXiv:2011.13230 (2020).
8. N. A. Gerasimenko, A. S. Chernyavsky, and M. A. Nikiforova, *ruSciBERT: A Transformer Language Model for Obtaining Semantic Embeddings of Scientific Texts in Russian.* — Doklady Mathematics **106**(suppl. 1) (2022), S95–S96. Available: https://doi.org/10.1134/S1064562422060072.
9. Sberbank, *GigaChat API: Overview*, 2024. Available: https://developers.sber.ru/docs/ru/gigachat/api/overview.
10. Z. Jiang, L. Gao, K. Yuan, Z. Gao, Z. Tang, and X. Liu, *Mathematics Content Understanding for Cyberlearning via Formula Evolution Map*, in: Proceedings of the ACM International Conference on Information and Knowledge Management (CIKM), 2018, pp. 37–46.
11. Y. Kuratov and M. Arkhipov, *Adaptation of Deep Bidirectional Multilingual Transformers for Russian Language*, ArXiv preprint arXiv:1905.07213 (2019).
12. OpenAI, J. Achiam, S. Adler, *et al.*, *GPT-4 Technical Report*, ArXiv preprint arXiv:2303.08774 (2024). Available: https://arxiv.org/abs/2303.08774.

13. S. Peng, K. Yuan, L. Gao, and Z. Tang, *MathBERT: A Pre-Trained Model for Mathematical Formula Understanding*, ArXiv preprint arXiv:2105.00377 (2021). Available: https://arxiv.org/abs/2105.00377.
14. M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer, *Deep Contextualized Word Representations*, ArXiv preprint arXiv:1802.05365 (2018).
15. S. Wang, Y. Guo, Y. Wang, H. Sun, and J. Huang, *SMILES-BERT: Large-Scale Unsupervised Pre-Training for Molecular Property Prediction*, in: Proceedings of the 10th ACM International Conference on Bioinformatics, Computational Biology and Health Informatics, 2019, pp. 429–436.
16. K. Yuan, L. Gao, Y. Wang, X. Yi, and Z. Tang, *A Mathematical Information Retrieval System Based on RankBoost*, in: Proceedings of the Joint Conference on Digital Libraries (JCDL), 2016, pp. 259–260.

Novosibirsk State University
E-mail: a.latushko@g.nsu.ru

Поступило 5 мая 2025 г.

Novosibirsk State University,
Institute of Informatics Systems SB RAS
E-mail: bruches@bk.ru