Computational Linguistics and Intellectual Technologies:
Proceedings of the International Conference "Dialogue 2025"

April 23–25, 2025

LORuGEC: the Linguistically Oriented Rule-annotated corpus for Grammatical Error Correction of Russian

Alexey Sorokin
MSU Institute for AI; Yandex
Lomonosov avenue, 27, corp. 1
a.sorokin at iai.msu.ru

Regina Nasyrova
MSU Institute for AI
r.nasyrova at iai.msu.ru

Abstract

We release LORuGEC – the first rule-annotated corpus for Russian grammatical error correction. The sentences in it are accompanied with the grammar rules governing their spelling. In total, we collected 48 rules with 348 sentences for validation and 612 for testing. LORuGEC appears to be challenging for open-source LLMs: the best F0.5-score is achieved by Qwen2.5-7B using two-stage finetuning and is only 50%. The closed YandexGPT4 Pro model achieves the score of 75%. Using a rule-informed retriever for fewshot example selection, we improve these scores up to 57% for Qwen and 81% for YandexGPT4 Pro.

Keywords: Grammatical Error Correction, Rule Annotation, Large Language Models, Fewshot Learning, Demonstration Selection.

DOI: 10.28995/2075-7182-2025-23-XX-XX

LORuGEC: лингвистически ориентированный корпус для задачи исправления грамматических ошибок в русском языке

Аннотация

Работа посвящена созданию корпуса LORuGEC – первого корпуса для исправления грамматических ошибок, в котором каждой ошибке сопоставлено правило, регулирующее корректное написание. Всего собрано 48 правил, корпус состоит из 348 валидационных примеров и 612 тестировочных. Из открытых языковых моделей среднего размера наилучшие результаты показывает модель Qwen2.5-7B, достигающая F0.5-меры в 50% при использовании двухступенчатого дообучения. Среди закрытых моделей лучше всех себя показывает модель YandexGPT4 Pro, чья F-мера при использовании fewshot равна 75%. При этом если использовать правиловую разметку для обучения ретривера, подбирающего демонстрационные примеры, то качество этих моделей можно улучшить до 57 и 81%, соответственно.

Ключевые слова: исправление грамматических ошибок, правиловая разметка, большие языковые модели, фьюшот-обучение, подбор демонстраций.

1 Introduction

Since the first works on Grammatical Error Correction (GEC), its primary application was for second language learning. When studying a foreign language, people tend to make multiple errors. That is why most of the corpora for GEC is based on foreign learners' texts or contain a mix of second (L2) and first (L1) language data. For example, in the case of English only the LOCNESS Corpus (Bryant et al., 2019) is based on both sources, while NUCLE(Dahlmeier et al., 2013) and Cambridge English Write&Improve Corpus (W&I)(Bryant et al., 2019) include only L2 data. The same holds for Russian, where both RULEC-GEC(Rozovskaya and Roth, 2019) and RU-Lang8(Trinh and Rozovskaya, 2021) consist of L2 and heritage data and only the recent GERA(Sorokin and Nasyrova, 2024) is based on native school texts.

As observed multiple times, L2 and L1 texts differ by the error distribution(Bryant et al., 2019; Flachs et al., 2020; Sorokin and Nasyrova, 2024). However, another factor that affects the complexity of grammatical errors is the source of the data. Most of the time, free-form essays serve as source texts for

GEC corpora. People tend to select expressions they are more confident in, reducing the risk of making grammatical errors, so some complex constructions may be underrepresented in GEC corpora. Thus, the models trained on such data have limited ability to collect complex errors, reducing their educational usefulness.

This observation was verified empirically by training large language models (LLMs) on GEC task using existing Russian corpora, such as GERA and RU-Lang8. We found that after such training, LLMs mostly improve precision, the change in recall is either less notable or even negative. Generally speaking, LLMs become more strict and less creative, which indicates that they excel in fixing "familiar" types of errors, which they were trained on, but mostly refrain from correcting other types – performing even worse than its basic version applied in a zero-shot mode.

Due to these considerations, our initial goal was to study the ability of large language models to correct complex grammatical errors. The current work primarily describes the first and the main part of this study – data collection. Our approach is case-oriented: we form a list of complex rules, using grammar handbooks as sources of data and then ask the annotators to collect sentence examples whose spelling is guided by these rules.

Eventually, we apply several models and approaches to the test sample of our data. As we expected, fine-tuning on existing corpora occurs to be suboptimal, even when the validation part of the collected data is included in the training data. In contrast, the best results were achieved by the few-shot approach. We also briefly discuss the algorithm of few-shot example selection that heavily affects the results. We hope that our data will become useful both for NLP and educational purposes. We make it freely available[1].

2 LORuGEC

We aimed at creating a Russian GEC dataset that would be more challenging for large language models (LLMs) and more linguistically-oriented, than existing corpora. LLMs are trained on large amounts of text data and therefore excel in extracting word cooccurrence patterns. Extracting such patterns is helpful in grammar-related tasks; for example, a model easily learns that the conjunction *что* requires a comma before it.

However, to implement some of rules of the Russian language, it is not enough to have profound knowledge of tokens and their co-occurrence, as they also require deeper understanding of semantics. Some of the challenges that models face are with common particles that may be written in one word or separately with the next token, depending on their meaning in a context (see Example 1), or commas that may be omitted in the sentences with several clauses (where commas, as a general rule, must be), if they have a common semantic component, for example, expressing place or time (Example 2).

(1) a. Он пошел **не** смотря вниз.

Он пошел не смотря вниз
He went not looking down

'He went without looking down.'

b. Он пошел **не**смотря на предупреждение.

Он пошел несмотря на предупреждение
He went not looking at warning

'He went despite the warning.'

(2) a. Они заполняли форму, и им приходило уведомление.

Они заполняли форму и им приходило уведомление
They filled form and to them came notification

'They filled the form, and a notification came to them.'

b. **Ранее** они заполняли форму и им приходило уведомление.

[1] `https://github.com/ReginaNasyrova/LORuGEC`

Ранее	*они*	*заполняли*	*форму*	*и*	*им*	*приходило*	*уведомление*
Earlier	they	filled	form	and	to them	came	notification

'Earlier they filled the form and [earlier] a notification came to them.'

2.1 Data

Rules of Russian grammar as well as examples to them were selected manually from the following grammar reference books, their electronic versions and educational websites (see more details on data extraction in the section below):

- High school Unified State Exam preparation books: (Berezina and Borisov, 2017) (Simakova, 2016)
- Academic handbook on spelling and punctuation: (Valgina et al., 2009), `http://orthographia.ru/`
- Handbook on the contemporary Russian language: (Valgina et al., 2002), `https://pedlib.ru/Books/6/0262/`
- Handbook on spelling and stylistics: (Rozental', 1997), `https://rosental-book.ru/`
- Dictionary of Russian collocations: (Kochneva, 1983)
- Educational web-sources: `https://orfogrammka.ru/`, `https://gramota.ru/biblioteka/spravochniki/`, `http://old-rozental.ru/`, `https://grammatika-rus.ru/`, `https://licey.net/free/4-russkii_yazyk/`, `https://www.yaklass.ru/p/russky-yazik/`

2.2 Collection and Annotation

Data were extracted and annotated by three bachelor students with a linguistic background, who are also Russian native speakers.[2] The annotators were given the following instruction:

- Select a source/sources of rules (see the list of chosen educational handbooks and websites in the section above), then choose several rules from different grammar sections: punctuation, spelling, grammar[3] and semantics. The authors of the paper also manually created a seed list of challenging rules, that the annotators were using as examples. During the annotation process, the selection of rules was additionally controlled by the paper authors.
- Find or construct 15 examples for each of the selected rules. As there is no available information on large language models' training data, to reduce the risk of compromising the dataset, several precautions were taken:
  - Preferably, choose sentences from different sources.
  - Avoid using quotations from fiction.
  - Refrain from selecting commonplace examples.
- Corrupt a fragment of each sentence which has to do with the rule it was selected for. If there may be a number of ways to make a mistake in a rule, it should be taken into account, while transforming the sentences for this rule. For instance, in Russian converbial clauses in the middle of the sentence must be marked with commas on both sides – Example 3 – so there are at least three ways of making a mistake: by overlooking the first comma, the second one or both.

  (3) Дети, гуляя по парку, ели мороженое.

Дети	*гуляя*	*по*	*парку*	*ели*	*мороженое*
Kids	walking	around	park	ate	ice cream

  'Kids ate ice cream, [while] walking around the park.'

  Our goal is to include diverse examples of errors for each rule, since it would more precisely reflect the set of possible mistakes in a text.
- For each rule test the YandexGPT3 Pro[4] model on the constructed sentences. If the model is imperfect in correcting the sentences, then try to generate several other sentences that belong to the same rule and present challenges to the model.

[2] The students earned credit hours as a result.

[3] The word *grammar* is polysemous, in GEC all kinds of errors in a text, except for the factual ones, are considered to be *grammatical errors*. Yet there are also specifically *grammatical errors*, which have to do with grammatical categories, e.g. wrong choice of number.

[4] `https://yandex.cloud/ru/docs/foundation-models/concepts/yandexgpt/models`

Since the source data creation requires targeted corruption of correct sentences, there is little chance for ambiguous corrections. In addition, a random subset of the annotated sentences was analyzed by the principal annotator, who found no discrepancies with the annotators‘ decisions.

2.3 Data Format

The dataset consists of rules, their definitions, information on their complexity for the YandexGPT Base model, pairs of corresponding tokenized[5] grammatical and ungrammatical sentences (see Table 1). There is some additional information, representing grammar sections which rules pertain to, sources of rules as well as indication of the subset for each sentence (validation or test, see more in the next section). There are few sentences in the dataset that do not contain any errors (see column *Correct source sentences* in Table 2), because it is also crucial to verify if models are prone to hypercorrection. These sentences are also marked with metadata.

The rule	**Did the base model have difficulties with the rule?**	**Initial sentence**	**Correct sentence**
Запятая перед союзом “как”: 2 [случай] Commas before the conjunction *kak*: second case	No	Иванова , как художника , я совсем не знаю . I don’t know Ivanov at all , as an artist.	Иванова как художника я совсем не знаю . I don’t know Ivanov at all as an artist.

Table 1: An example of rule from the dataset. Some metadata as well as other sentences for this rule were omitted for illustrative purposes.

We also present our data in .M2, which is a conventional GEC format. According to the .M2-standard, the source text is denoted with `S`, while the corresponding edits are prefixed with `A`. Each edit consists of the error span, error type, correction, if the edit is optional or required, additional remarks and annotator ID, yet we do not make use of error types:

S Иванова , как художника , я совсем не знаю .
A 1 2|||None||||||REQUIRED|||-NONE-|||0
A 4 5|||None||||||REQUIRED|||-NONE-|||0

2.4 Rules Description and Statistics

We gathered 48 rules from 4 grammar sections. The majority of them represent punctuation and spelling:

- **Grammar**
  - 1 Incorrect expression of government
  - 2 Declension of cardinal numerals
  - 3 Declension of numerals *poltora*, *poltory*, *poltorasta*
  - 4 Agreement between the participle and the word it defines
- **Punctuation**
  - 5 Commas in idiomatic expressions
  - 6 Commas between homogeneous subordinate clauses
  - 7 Commas between subordinate and main clauses
  - 8 Commas between the two conjunctions
  - 9-11 Commas before the conjunction *kak*: 3 instances
  - 12 Sentences with homogeneous parts
  - 13 Converbs after conjunctions

[5]We made use of NLTK Tokenizer: `https://www.nltk.org/api/nltk.tokenize.html`.

14 Clauses related to the personal pronoun
15 Clauses that are distant from the word they define
16 Punctuation in meaningful (indecomposable) expressions
17 Linking words and constructions
18 Recurring conjunctions
19 Dashes in sentences with no conjunctions
20 Dashes between the subject and the predicate
21 Dashes in case of appositions

- **Semantics**
  22 Collocations
  23 Pleonasms
- **Spelling**
  24 *n* and *nn* in the suffixes of adjectives
  25 Vowels in the suffixes of participles
  26 Noun suffixes *on'k*, *en'k*
  27 Suffixes *ic*, *ec* in neuter nouns
  28 Suffixes *ek*, *ik*
  29 Adjective suffixes *insk*, *ensk*
  30 Prefixes *pre* and *pri*
  31 *y* and *i* after prefixes
  32 Vowels after *c*
  33 Vowels after sibilants
  34 Separating letters[6]
  35 Hyphens as part of written equivalents of complex words
  36 Joint, separate or hyphenated spelling of adverbs
  37 Compound adjectives
  38 Particle *taki*
  39 *zato*
  40 *ottogo*
  41 *prichyom* and *pritom*
  42 *takzhe*
  43 *chtoby*
  44 *pol-*
  45 *ne* with verbs
  46 *ne* with adjectives
  47 *ne* with participles
  48 *ne* with nouns

Our research during the annotation showed that 29 out of 48 collected rules were challenging for the YandexGPT3 Pro. As may be observed on the Figure 1, the largest percentages of collected complex rules occur among punctuation and semantics. This partly proves our hypothesis that rules which require the understanding of semantics pose a more serious challenge to LLMs.

We collected 960 pairs of sentences, which were split into validation and test subsets so that for each rule at least 9 sentences or approximately two thirds of collected sentences would be allocated to the test partition (see Figure 2). Consequently, the size of the test subset is twice as large as the size of the validation one (see Table 2). Additionaly, unlike the latter, only the test subset includes initially correct sentences (for hypercorrection considerations). In both samples, however, two thirds of the sentences come from complex rules.

[6] ь, ъ are implied here, as they do not have literal ways of transliteration

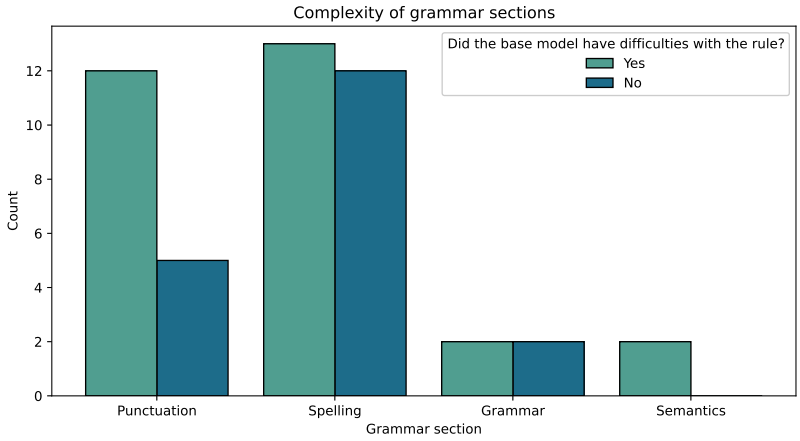

Figure 1: Complexity of different grammar sections is expressed by the number of complex for the YandexGPT3 Pro model rules. We considered the rule to be difficult if the model failed to correct some of its sentences (see 2.2).

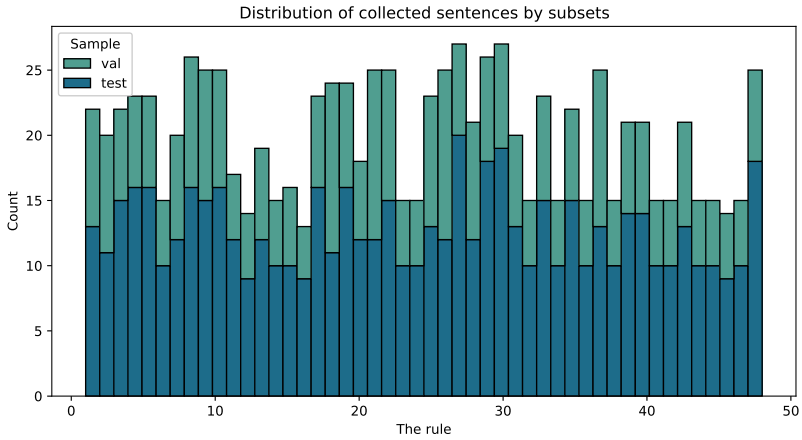

Figure 2: Distribution of sentences for each rule among validation and test samples.

Sample	**Sentences**	**Correct source sentences**	**Sentences for complex rules (%)**	**Tokens**
Validation	348	0	250 (71.84)	5,579
Test	612	31	419 (68.46)	10,131

Table 2: Statistics on the validation and test samples of LORuGEC.

2.5 Comparison to other corpora

Here we compare our corpus to existing corpora for Grammatical Error Correction of Russian: RULEC-GEC(Rozovskaya and Roth, 2019), RU-Lang8(Trinh and Rozovskaya, 2021) and GERA(Sorokin and Nasyrova, 2024). It differs from them in several aspects:

- To the best of our knowledge, that is the only GEC corpus where all the errors are matched with corresponding grammar rules.
- Our corpus is purposely created for evaluation purposes, not for training. Therefore, it has no

Sample	Sentences	Tokens
RULEC-GEC	12,480	206,258
RU-Lang8	4,412	54,741
GERA	6,681	119,068
LORuGEC	960	15,710

Table 3: Statistics on the validation and test samples of our dataset.

training subset and is much smaller than other corpora. On the other hand, almost all sentences of our corpus contain errors and are supposed to be challenging in contrast to other GEC data.

- Since corpus examples were created via corruption, for the vast majority of mistakes there is only one possible correction, increasing the trustworthiness of evaluation scores.

3 Model evaluation

Our preliminary studies demonstrated that large language models (LLMs) achieve state-of-the-art (SOTA) or near-SOTA results on the existing Russian GEC corpora, outperforming other methods. Therefore we restricted our attention to decoder-based LLMs. In the first series of our experiments we evaluate several open-source models as well as the closed YandexGPT model[7]. We prefer YandexGPT to GPT4 and other API-based large models since Russian was the main language during its training. Between the open-source models we select the multilingual Qwen2.5-3B Instruct[8], Qwen2.5-7B Instruct[9] (Yang et al., 2024) and the T-Lite 7B model[10], which is also based on Qwen. We selected these models among other variants as during preliminary experiments they showed a decent ability to correct grammatical errors in zero-shot mode, outperforming other open-source models, such as LLaMA or Mistral. Since our goal is to investigate the ability of LLMs to perform error correction task without finetuning, we selected the instruction-based versions of Qwen models. Additionally, these models show state-of-the-art performance on other Russian GEC datasets, outperforming other approaches.

In the first series of experiments we report the results of 0-shot, 1-shot and 5-shot runs. The demonstrations for fewshot are selected at random. We also evaluate finetuned versions of open-source models. Since the validation part of our corpus is rather small, we compare two variants of finetuning:

1. Train the models on the concatenation of available Russian GEC corpora: RULEC-GEC, RuLang8 and GERA.
2. Further tune the model on the validation part of the LORuGEC corpus.

As it is commonly done, we score the tokenized model outputs with M2scorer(Dahlmeier et al., 2013) and report precision, recall and F0.5 score, using F0.5 as the main metric.

The first result of our work is the difference between closed-source and open-source models. A partial explanation is the larger size of YandexGPT Pro model, however, the Lite model also clearly outperforms the open-source models. We have two possible explanations: first, many examples are taken from the school textbooks that likely were in the training data of Yandex models. Second, open-source LLMs are aligned on "creative" instruction-following tasks that require the model to rewrite the input text significantly. This makes them prone to overcorrection and hallucination that explains their moderate precision in comparison to recall.

Concerning open-source LLMs, we also observe a clear difference between the behaviour of the basic and finetuned models. The finetuned models follow the pattern of traditional GEC models based on smaller LLMs or Transformer networks as their precision is much higher than recall. Conversely, the

[7] `https://yandex.cloud/ru/docs/foundation-models/concepts/yandexgpt/models`, assessed 20th January, 2025.

[8] `https://huggingface.co/Qwen/Qwen2.5-3B-Instruct`

[9] `https://huggingface.co/Qwen/Qwen2.5-7B-Instruct`

[10] `https://huggingface.co/t-tech/T-lite-it-1.0qwen`

Model	0-shot	1-shot	5-shot	FT	FT+LORuGEC
Qwen-3B	29.4/33.9/30.2	29.9/31.1/30.1	34.8/33.1/34.4	45.1/18.6/35.1	48.4/32.1/44.0
Qwen-7B	38.6/39.8/38.9	43.5/38.4/42.4	46.7/39.5/45.0	50.8/18.4/37.6	54.2/39.8/50.5
T-Lite	31.9/48.5/34.3	37.9/43.7/38.9	41.8/44.5/42.3	54.1/22.4/42.1	61.0/46.5/57.4
YaGPT4 Lite	64.0/68.5/64.8	68.3/70.8/68.8	67.4/69.8/67.9	NA	NA
YaGPT4 Pro	68.6/72.8/69.4	72.5/71.0/72.2	75.2/73.6/74.9	NA	NA

Table 4: Comparison of different LLMs on the test set in zero-shot, fewshot and finetuning (FT) mode. FT+LORuGEC denotes two-stage funing with initial training on 3 basic Russian GEC corpora and further finetuning on the validation set of LORuGEC. We report precision, recall and F0.5 score (the main metric), separated by slashes.

pretrained models without finetuning tend to overcorrect, not only correcting grammar, but also trying to improve sentence fluency or make it more "standardized". This results in decent recall but poor precision. We suppose the alignment procedure of modern instruction-tuned LLMs to be the reason, since traditional alignment datasets contain a significant fraction of text editing tasks that require more extensive rewriting, than GEC. Additionally, the T-Lite model often fails to follow the prompt precisely, adding superfluous explanations or comments, but these format violation errors are nearly avoided with fewshot demonstrations.

To unveil the potential of fewshot learning and the usefulness of additional linguistic information, we perform a second series of experiments. We try to expose the models with the most relevant fewshot information as possible. The simplest solution could be to select the demonstrations from the same rule subset as the sentence under consideration. However, for arbitrary sentences their rule labels are not available in practice. Since the rule set is open and no corpus can cover all the grammatical rules of the language, training a rule classifier is also not a complete solution.

Our approach is to equip the demonstration selection algorithm with a similarity model. Common sentence embedding models mostly reflect semantic similarity that is irrelevant for our task. We hypothesize that grammatical similarity might be reflected by an encoder model trained on grammar-related task. To verify this hypothesis, we train an analogue of GECTOR model(Omelianchuk et al., 2020). This approach reduces grammatical error correction to sequence labeling, with labels encoding elementary edits operations. Thus, similar states of GECTOR encoder correspond to similar edit operations and, consequently, to similar grammatical rules. We train the GECTOR model on Russian grammatical error data, using the three available corpora (RULEC-GEC, RU-Lang8 and GERA) as well as up to 1M sentences with synthetic errors. During preliminary experiments we found that a single vector representation for a sentence is not enough and decided to represent a sentence by up to 3 states, corresponding to the most probable error positions.

We further tune GECTOR embedder on the task of rule classification using contrastive learning. To elaborate, for every source sentence in the validation set we search for its closest neighbour in embedding space and compel this neighbour to belong to the same class. Formally, for every embedding h we minimize the conventional triplet loss:

$$L(h, h^+, h^-) = \max(\rho(h, h^+) - \rho(h, h^-) + d, 0),$$

where h^+ is the closest example belonging to the same rule class and h^- is the closest neighbour from another class. Finetuning is performed on the validation set of our corpus.

The comparison of different embedding strategies is provided in Table 5.

We observe that GECTOR-based demonstration selection consistently outperforms the random one. This shows that encoder-based GEC models actually encode information about grammatical rules in its hidden states even when training on external GEC data. Tuning the encoder on in-domain data further

Model	k	random	GECTOR	GECTOR+ft
Qwen2.5-7B	1	43.5/38.4/42.4	47.6/42.9/46.6	**49.1/46.5/48.5**
	5	46.7/39.5/45.0	53.3/48.1/52.1	**58.9/52.9/57.6**
YandexGPT4 Lite	1	68.3/70.8/68.8	71.0/72.8/71.3	**73.4/74.4/73.6**
	5	67.4/69.8/67.9	70.9/71.6/71.0	**76.3/73.1/75.6**
YandexGPT4Pro	1	72.5/71.0/72.2	74.3/75.9/74.6	**78.5/76.9/78.2**
	5	75.2/73.6/74.9	80.4/75.4/79.3	**82.5/77.8/81.5**

Table 5: Comparison of random (*random*), embedder-based(*GECTOR*) and finetuned embedder-based (*GECTOR+ft*) for several LLMs. Best results across selection methods are in bold. We report precision, recall and F0.5 score (the main metric), separated by slashes.

improves the quality of demonstrations. Note that contrastive training requires rule labels, thus rule type annotation of our corpus is useful not only for linguistic, but also for practical purposes.

4 Conclusion

We created a linguistically-oriented evaluation corpus for Grammatical Error Correction of Russian. It appears to be challenging to current open-source models both in zero-shot mode or after finetuning on other Russian GEC corpora. However, the closed YandexGPT Pro4 model yields much higher scores, achieving the F0.5 score of 69% in zero-shot mode and 74% with 5-shot.

Since our corpus is additionally equipped with rule type information, we also show the utility of this annotation by training an encoder to assign similar vectors for examples with analogous mistakes. Using the trained encoder to select similar examples, we improve the quality of 5-shot error correction up to 81%. We hope that our study will shed additional light on the role of linguistic information in grammatical error correction and provide further insight for investigating in-context learning methods.

5 Acknowledgements

We thank Yandex and especially Marina Kosheleva for providing a grant for access to YandexGPT.

References

Svetlana Berezina and Nikolaj Borisov. 2017. *Russkij yazyk v sxemax i tablicax*. Eksmo, Moskva.

Christopher Bryant, Mariano Felice, Øistein E Andersen, and Ted Briscoe. 2019. // *Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications*, P 52–75.

Daniel Dahlmeier, Hwee Tou Ng, and Siew Mei Wu. 2013. Building a large annotated corpus of learner English: The NUS corpus of learner English. // Joel Tetreault, Jill Burstein, and Claudia Leacock, *Proceedings of the Eighth Workshop on Innovative Use of NLP for Building Educational Applications*, P 22–31, Atlanta, Georgia, June. Association for Computational Linguistics.

Simon Flachs, Ophélie Lacroix, Helen Yannakoudakis, Marek Rei, and Anders Søgaard. 2020. Grammatical error correction in low error density domains: A new benchmark and analyses. // *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, P 8467–8478.

Elena Kochneva. 1983. *Slovar' sochetaemosti slov russkogo yazyka*. Russkij yazyk, Moskva.

Kostiantyn Omelianchuk, Vitaliy Atrasevych, Artem Chernodub, and Oleksandr Skurzhanskyi. 2020. GECToR – grammatical error correction: Tag, not rewrite. // Jill Burstein, Ekaterina Kochmar, Claudia Leacock, Nitin Madnani, Ildikó Pilán, Helen Yannakoudakis, and Torsten Zesch, *Proceedings of the Fifteenth Workshop on Innovative Use of NLP for Building Educational Applications*, P 163–170, Seattle, WA, USA → Online, July. Association for Computational Linguistics.

Ditmar Rozental'. 1997. *Spravochnik po pravopisaniyu i stilistike*. Komplekt, SPB.

Alla Rozovskaya and Dan Roth. 2019. Grammar error correction in morphologically rich languages: The case of Russian. *Transactions of the Association for Computational Linguistics*, 7:1–17.

Elena Simakova. 2016. *Russkij yazyk: Novyj polnyj spravochnik dlya podgotovki k EGE'*. AST:Astrel', Moskva.

Alexey Sorokin and Regina Nasyrova. 2024. GERA: a corpus of Russian school texts annotated for Grammatical Error Correction. *// Proceedings of The 12th International Conference on Analysis of Images, Social Networks and Texts, to appear*.

Viet Anh Trinh and Alla Rozovskaya. 2021. New dataset and strong baselines for the grammatical error correction of Russian. *// Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, P 4103–4111.

Nina Valgina, Ditmar Rozental', and Margarita Fomina. 2002. *Sovremennyj russkij yazyk: Uchebnik*. Logos, Moskva.

Nina Valgina, Nataliya Es'kova, Ol'ga Ivanova, Svetlana Kuz'mina, Vladimir Lopatin, and Lyudmila Chel'cova. 2009. *Pravila russkoj orfografii i punktuacii. Polnyj akademicheskij spravochnik*. AST, Moskva.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, et al. 2024. Qwen2. 5 technical report. *arXiv preprint arXiv:2412.15115*.