ПРАВИТЕЛЬСТВО РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное образовательное учреждение высшего образования
Национальный исследовательский университет
«Высшая школа экономики»

Факультет гуманитарных наук
Образовательная программа
«Фундаментальная и компьютерная лингвистика»

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА

На тему «Исследование возможностей больших языковых моделей в оценке лингвистической приемлемости русских предложений»

Тема на английском “Studying Large Language Models' Capabilities of Judging Linguistic Acceptability on Russian Sentences”

Студентка 4 курса
группы № БКЛ-201
Ткач Анна Сергеевна

Научный руководитель
Клышинский Эдуард Станиславович
доцент Школы лингвистики ФГН

Москва, 2024

Abstract

In recent years, Large Language Models (LLMs) have demonstrated impressive performance on various Natural Language Processing (NLP) tasks. However, little is still known about whether these models have any language knowledge. In this study, we aimed to investigate whether recent Russian LLMs possess any grammatical knowledge and, therefore, can be used as assistants while teaching grammar to L2 learners. We have tested two Russian LLMs (Gigachat and YandexGPT) on their ability to distinguish grammatical sentences from ungrammatical ones, categorise the types of errors, correct errors and provide explanations of errors. For this purpose, we have utilised the RuCoLA (Russian Corpus of Linguistic Acceptability) dataset. By using multiple prompts, we have run each of the models with 8 scenarios, which include different tasks. Our results show that the performance of the LLMs differ both from model to model and from task to task. We suggest that the models are used in the fields of grammar teaching or correction to a limited degree, since they do not always demonstrate sufficient knowledge of grammar of the Russian language. Finally, we also present our own labelling scheme, which can be used for human assessing of error explanations and corrections given by models.

Contents

1. Introduction

In recent years, Large Language Models (LLMs), such as LLaMA, BLOOM or Mistral, have taken over the field of Natural Language Processing (NLP). It is well known that such models can perform with high results on multiple NLP tasks, such as question answering, text generation, machine translation, etc. A lot of studies have provided evidence of good performance of LLMs' on different tasks (Bang et al., 2023; Hendy et al., 2023; Kocoń et al., 2023).

Since LLMs are able to perform multiple language tasks, it could be promising to use these models while teaching or correcting grammar. For example, it would be useful to have a chatbot capable of distinguishing grammatical errors, explaining and correcting them. However, there is still little evidence about whether (and to what extent) do (large) language models possess actual language knowledge.

In this study, we aim to test whether Russian LLMs have sufficient knowledge of the Russian grammar to be used as assistants while teaching or studying Russian. Two main objectives of our study are the following: (1) to test if these LLMs' are capable of giving linguistic acceptability judgements for Russian sentences, and (2) to examine whether the LLMs' are able to provide comprehensible explanations for their judgements. Moreover, we want to figure out which scenarios of prompting can be more efficient for solving different tasks connected with grammar (e.g., labelling sentences as (in)correct or explaining errors).

To examine Russian LLMs' grammar knowledge, we use the Russian Corpus of Linguistic Acceptability dataset (RuCoLA). We evaluate two recently developed models: GigaChat and YandexGPT. Using different prompting strategies, we run experiments with several tasks and their combinations, which test the LLMs' ability to judge sentences as (un)acceptable, to define the error type, to explain and to correct the error. In this study, the models' results will be evaluated using common metrics (accuracy and Matthews correlation coefficient), as well as manually created schema for assessing the quality of the models' explanations and corrections.

This thesis has the following structure: the next section is the Related work, where we briefly discuss recent approaches and papers in the field of testing neural models for grammar knowledge, as well as some theoretical background. In the “Materials and Methods” part, we present the dataset we used to evaluate the models, the models tested, the design of experiments and the methods of evaluating the models’ performance. In the next two sections, the obtained results are described and discussed. Finally, the last section of the thesis is the Conclusion.

All the code used for experiments and annotation of data is located in the github repository, link to which can be found in Appendix A.

2. Related work

2.1 Previous research of language knowledge in LLMs

Since the LLMs have taken over the field of NLP, in recent years there have been some papers which studied the degree of language knowledge LLMs possess.

For example, Ortega-Martín et al. in their paper ‘Linguistic ambiguity analysis in ChatGPT’ examine whether (and to what extent) the LLM is capable of handling multiple types of ambiguity (e.g., lexical or syntactic) using a prompting method. The results demonstrate that some categories of ambiguity (e.g., semantic) are more simple for ChatGPT than others.

Another study which deals with linguistic competence of LLMs is the recent paper[1] by Dentella et al. With the help of multiple prompts, the authors ask GPT-3 to perform grammaticality judgement tasks and answer comprehension questions concerning some uncommon language phenomena. The results of this study show that the model is not able to provide proper grammatical judgments and explanations.

Concerning the ability of LLMs to explain grammar, this competence of models was examined in ‘GEE! Grammar Error Explanation with Large Language Models’ written by Song et al. The authors of the paper propose a new task – Grammar Error

[1] Dentella, V., Murphy, E., Marcus, G., & Leivada, E. (2023). Testing AI performance on less frequent aspects of language reveals insensitivity to underlying meaning. arXiv preprint arXiv:2302.12313.

Explanation and assess the performance of GPT-4 on this task. The results show that the model provides successful explanations only in 60% of cases.

2.2 Acceptability Judgments in Linguistics

The concept of acceptability judgments was first introduced by Chomsky in his *Syntactic Structures*. It's obvious, then, that this approach initially became applied in generative syntax theory (for example, (Schütze, 2016) comments on a plethora of studies in 1980-s which used judgments to construct grammatical theories within the Generative Theory field).

Later, the method of eliciting acceptability judgments has become widely used in other fields of linguistics, for example, semantics or morphosyntax. See, for instance, (De Villiers, P. A. and de Villiers, J. G, 1972) and (Bermel and Knittl, 2012).

As acceptability judgments became more popular, various approaches to elicit them appeared. These include, for instance, binary rating (yes-no), Likert-scale rating, forced-choice between minimal pairs and magnitude estimation. The overview of these approaches can be found in (Sprouse, 2018). Based on these approaches, multiple datasets were created.

2.3 Linguistic Acceptability in NLP

Recently, two approaches have become common to evaluate language models' knowledge of grammar: the minimal-pairs approach and the yes-no one (binary classification). Minimal-pairs method represents the forced choice between minimal pairs (acceptable and unacceptable sentences), which was first introduced in (Warstadt et al., 2020). The goal for a model is to assign a higher possibility to an acceptable sentence than to an unacceptable one.

In this thesis, experiments are based on the second approach, which is to elicit and evaluate acceptability judgments from models via binary classification. Here, the goal of a model is to predict the class (correct/incorrect) of a given sentence.

The first dataset for this task was CoLA (Corpus of Linguistic Acceptability), created in 2019. It was described in the paper 'Neural network acceptability judgments' by Warstadt et al. In this paper, the authors presented the dataset consisting of approximately 10k (un)grammatical English sentences and examined the performance of some RNNs (recurrent neural networks) on this task. They conclude that the performance of RNNs is not comparable with that of humans.

Since the creation of CoLA its analogues for multiple other languages were introduced. For instance, there is CoLAC (for Chinese), ItaCoLA (for Italian), DaLAJ (for Swedish). Moreover, recently, the multilingual benchmark on linguistic acceptability, MELA, was presented. It consists of 48k sentences written in 10 languages.

The linguistic acceptability dataset for Russian is RuCoLA, designed by Mikhailov et al. This dataset consists of more than 13k Russian sentences, both acceptable and unacceptable. There are both in-domain sentences (taken from specialised literature) and out-of-domain ones (generated by neural models). The information of errors type and sources of sentences is also provided. The authors of the dataset have conducted several experiments with multiple baseline models (including non-neural models and fine-tuned transformer ones) and conclude that the performance of models is not sufficient, compared to humans'.

Concerning LLMs, various studies used CoLA-style datasets to examine the linguistic competence of models. For instance, in the papers "ChatGPT: Jack of all trades, master of none" [2]and "Can chatgpt understand too? A comparative study on chatgpt and fine-tuned bert"[3] researchers compared the performance of ChatGPT on this task with other SOTA models (e.g., BERT) and stated that the models showed comparable language knowledge.

2.4 Prompting

A prompt is a textual instruction to a language model written in natural language. The formal description of the concept 'prompt' was introduced in (Liu et al., 2023). In this paper, the authors oppose the traditional supervised learning and prompting. In the first case, a model predicts an output based on the received input by modelling a

[2] Kocoń, J., Cichecki, I., Kaszyca, O., Kochanek, M., Szydło, D., Baran, J., ... & Kazienko, P. (2023). ChatGPT: Jack of all trades, master of none. Information Fusion, 99, 101861.

[3] Zhong, Q., Ding, L., Liu, J., Du, B., & Tao, D. (2023). Can chatgpt understand too? a comparative study on chatgpt and fine-tuned bert. arXiv preprint arXiv:2302.10198.

probability P(y|x; θ), where θ stands for the models' parameters. Conversely, in the case of prompting the prediction of output is based on the modelling of probability of input itself. Furthermore, this paper suggests some classification of prompts: for example, prompts can be divided into cloze and prefix or into discrete and continuous. Some techniques of automated prompt generation are also suggested.

Since then, multiple papers suggested various prompting techniques for improving models' performance. In this thesis, we use two of them, which are providing a model with an example and assigning a model a particular role (in our case, a teacher of the Russian language). The first technique is called few-shot learning and was first described in the paper by Brown et al. The latter method, role and context prompting, was suggested to be beneficial for LLMs' performance for various tasks, for instance, see (Santu et al., 2023) and (Rong et al., 2023).

3. Materials and Methods

3.1 Dataset

For our experiments, we use the RuCoLA[4] (Russian Corpus of Linguistic Acceptability) dataset, which is a benchmark designed for evaluating the linguistic competence of Russian language models. RuCoLA is the analogue of the CoLA[5] (Corpus of Linguistic Acceptability) dataset for the Russian language.

RuCoLA contains sentences labelled as acceptable or unacceptable. In addition to it, the category of error is marked. There are four possible types of errors: Morphology, Syntax, Semantics or Hallucination. Table 1 provides examples of sentences from the dataset, which contain errors of various categories. In the case of labels, '0' stands for an incorrect sentence, and '1' is a correct sentence.

Sentence	Label	Category
Легко показать, что всякий иной путь длиньше этого.	0	Morphology
Идя вдоль берега, морской воздух приятно освежал наши лица.	0	Syntax
Ты когда-нибудь находился в Казани?	0	Semantics

[4] https://rucola-benchmark.com/

[5] https://nyu-mll.github.io/CoLA/

Случаи пива вылетели на улицу.	0	Hallucination
В этом дне была какая-то особенная прелесть.	1	0

Table 1. Examples from the RuCoLA dataset.

We run our experiments with the validation part of the dataset, which consists of 2.7k sentences.

3.2 Models

We evaluate two recent LLMs for the Russian language: GigaChat and YandexGPT. They are both conversational GPT-like models, developed by Yandex and Sber, respectively.

At the initial stage, we also tried to test Saiga, which is a Russian chatbot based on LLaMA-2. However, we found that it was too incompetent, so we decided not to conduct further analysis of it.

3.3 Prompting

We use eight different scenarios to assess the models. There are four tasks which could be asked to perform: label (we prompt model to predict the class of the sentence: acceptable or not), category (we prompt model to predict the type of error), explanation (we prompt model to explain why the sentence is incorrect), correction (we prompt model to correct an unacceptable sentence). We combine them in the following way: label, label+category, label+explanation, label+correction, label+category+explanation, label+category+correction, label+explanation+correction, label+category+explanation+ correction. For example, in the ‘label+category+correction’ scenario, we prompt the models to mark a sentence as (in)correct, and in case it is incorrect, to give the type of error and to correct the error.

Also, we run each scenario five times with each model in order to take into account the possible instability of the LLMs’ responses. For each scenario, the models receive all sentences from the dataset (one at a time), each accompanied by an instruction.

First, in all scenarios we use system prompts to give a context to a model and assign it a role. The system prompt is equal for all the scenarios for both models: “*Ты* *учитель русского языка. Твоя задача — помогать своим ученикам.*”. We chose to use this system prompt based on the results of the label scenario obtained with and without it.

For each model, we ran the Label task 3 times with assigning a teacher role and without doing it. The models’ performance on these runs (measured as accuracy) can be seen in the following table.

	With role	Without role
YandexGPT	0.7888±0.0002	0.7629±0.0008
GigaChat	0,7525±0	0.7515±0.0016

Table 2. Accuracy of YandexGPT and GigaChat with and without assigning a teacher role.

As the results with the system prompt assigning a teacher role are better, we decided to use this system prompt in all scenarios for both models.

All the prompts used for YandexGPT and GigaChat are presented in the Table 2 below (the system prompt is not presented here). The prompts for the two models are rather similar, except some minor details. For example, in some scenarios we ‘ask’ YandexGPT to ‘check the sentence very carefully’ as it tends to ignore most of the errors otherwise.

	YandexGPT	GigaChat
Label	'Правильно ли следующее предложение с точки зрения русского языка: "{input}"? Ответь только одно слово: да или нет.'	Правильно ли следующее предложение с точки зрения русского языка: "{input}"? Ответь только "да" или "нет".
Label, category	Есть ли ошибка в предложении: "{input}"? Если ошибки нет, ответь только 1 слово "нет" (пожалуйста, не добавляй ничего), если ошибка есть – обязательно укажи также тип ошибки: синтаксис, морфология,	Есть ли ошибка в предложении: "{input}"? Если ошибки нет, ответь только "нет". Если ошибка есть – ответь да и укажи, к какому типу относится ошибка. Ошибка может относиться к одному из следующих

	семантика или галлюцинации. Примеры ошибочных предложений: морфология — "Увидал, что у санитара нет сапогов — велел дать.", синтаксис — "Этим летом не никуда ездили.", семантика — "Он живет в небольшом городке у Волги.", галлюцинации — "Люк останавливает удачу от этого.". Пожалуйста, внимательно проверь предложение.	типов: морфологическая, синтаксическая, семантическая или галлюцинация. Примеры ошибок: морфологическая — "Увидал, что у санитара нет сапогов — велел дать." (правильный вариант — "Увидал, что у санитара нет сапог — велел дать."), синтаксическая — "Этим летом не никуда ездили." (правильный вариант — "Этим летом никуда не ездили"), семантическая — "Он живет в небольшом городке у Волги.", галлюцинация — "Люк останавливает удачу от этого.".
Label, explanation	Перед тобой предложение на русском языке, в котором, возможно, есть ошибка: {input}. Ответь, есть ли ошибка в этом предложении. Если предложение неправильное, объясни, в чем ошибка. Объясни ошибку подробно, пожалуйста.	'Ты учитель русского языка. Перед тобой предложение на русском языке, в котором, возможно, есть грамматическая ошибка: {input}. Ответь, правильное ли это предложение. Если предложение неправильное, объясни, в чем ошибка.'
Label, correction	Перед тобой предложение на русском языке, в котором, возможно, есть ошибка: {input}. Если в предложении есть ошибка, исправь предложение и напиши правильный вариант. Если предложение не содержит ошибок, ответь "в предложении нет ошибок". Пожалуйста, проверяй предложение очень внимательно.	Дано предложение на русском языке, в котором, возможно, есть ошибка: {input}. Если данное предложение неправильное, напиши "есть ошибка" и исправь предложение. Если данное предложение правильное, напиши "ошибок нет".
Label, category, explanation	Перед тобой предложение на русском языке, в котором, возможно, есть ошибка: {input}. Если в предложении есть ошибка, ответь, к какой категории она относится, и объясни, в чем заключается ошибка. Ошибка может относиться к одной из следующих категорий: морфология, синтаксис, семантика или галлюцинации. В ответе может быть только одна категория ошибки, нельзя приводить несколько. Примеры ошибочных предложений: синтаксис — "Этим летом не никуда ездили.", семантика — "Он живет в небольшом городке у Волги.", морфология — "Увидал, что у санитара нет сапогов — велел	Перед тобой предложение на русском языке, в котором, возможно, есть ошибка: {input}. Если в предложении есть ошибка, ответь, к какой категории она относится, и объясни, в чем заключается ошибка. Ошибка может относиться к одной из следующих категорий: морфологическая, синтаксическая, семантическая или галлюцинации. Примеры ошибочных предложений: синтаксическая ошибка — "Этим летом не никуда ездили.", семантическая ошибка — "Он живет в небольшом городке у Волги.", морфологическая ошибка — "Увидал, что у санитара нет сапогов — велел дать.", галлюцинация — "Люк

	дать.", галлюцинации — "Люк останавливает удачу от этого.". Напиши ответ в формате: категория ошибки:, объяснение: [твое объяснение]. Если предложение не содержит ошибок, ответь "в предложении нет ошибок". Пожалуйста, проверяй предложение очень внимательно..	останавливает удачу от этого.". Напиши ответ в формате: категория ошибки:, объяснение: [твое объяснение]. Если предложение не содержит ошибок, ответь только "в предложении нет ошибок" и больше ничего.
Label, category, correction	Перед тобой предложение на русском языке, в котором, возможно, есть ошибка: {input}. Если в предложении есть ошибка, ответь, к какой категории она относится, и напиши исправленное предложение. Не объясняй свое решение. Ошибка может относиться к одной из следующих категорий: морфологическая, синтаксическая, семантическая или галлюцинации. Примеры ошибочных предложений: синтаксическая ошибка — "Этим летом не никуда ездили.", семантическая ошибка — "Он живет в небольшом городке у Волги.", морфологическая ошибка — "Увидал, что у санитара нет сапогов — велел дать.", галлюцинация — "Люк останавливает удачу от этого.". Напиши ответ в формате: категория ошибки: [синтаксическая/семантическая/морфологическая/галлюцинация], исправленное предложение: [правильное предложение]. Если предложение не содержит ошибок, ответь только "в предложении нет ошибок". Пожалуйста, проверь предложение очень внимательно.	Перед тобой предложение на русском языке, в котором, возможно, есть ошибка: {input}. Если в предложении нет ошибок, ответь только "нет ошибок". Если в предложении есть ошибка, ответь, к какой категории она относится: морфологическая, синтаксическая, семантическая или галлюцинация. Также напиши, как можно исправить предложение. В этом случае напиши ответ в формате: "категория: [категория ошибки], правильный вариант: [правильный вариант]". Примеры ошибочных предложений: синтаксическая ошибка — "Этим летом не никуда ездили.", семантическая ошибка — "Он живет в небольшом городке у Волги.", морфологическая ошибка — "Увидал, что у санитара нет сапогов — велел дать.", галлюцинация — "Люк останавливает удачу от этого.".
Label, explanation, correction	Перед тобой предложение на русском языке, в котором, возможно, есть ошибка: {input}. Если в предложении есть ошибка, объясни, в чем именно ошибка, и исправь предложение. Если ты нашел ошибку, напиши ответ в формате: объяснение: [твое объяснение], исправленное предложение: [правильное предложение]. Если предложение не содержит ошибок, ответь только	Перед тобой предложение на русском языке, в котором, возможно, есть ошибка: {input}. Если в предложении есть ошибка, подробно объясни, в чем именно заключается ошибка, и исправь предложение. Если ты нашел ошибку, напиши ответ в формате: "объяснение: [твое объяснение], исправленное предложение: [правильное предложение]". Если предложение не содержит

	"в предложении нет ошибок". Пожалуйста, если ты нашел ошибку, опиши ее максимально подробно, чтобы помочь своему ученику.	ошибок, ответь только "в предложении нет ошибок". Пожалуйста, если ты нашел ошибку, опиши ее максимально подробно, чтобы помочь своему ученику.
Label, category, explanation, correction	Перед тобой предложение на русском языке, в котором, возможно, есть ошибка: {input}. Если в предложении есть ошибка, ответь, к какой категории она относится. Также объясни, в чем именно ошибка, и исправь предложение. Если предложение не содержит ошибок, ответь только "в предложении нет ошибок". Если ты нашел ошибку, напиши ответ в формате: категория: [категория ошибки], объяснение: [твое объяснение], исправленное предложение: [правильное предложение]. Ошибка может относиться к одной из следующих категорий: морфологическая, синтаксическая, семантическая или галлюцинации. Примеры ошибочных предложений: синтаксическая ошибка — "Этим летом не никуда ездили.", семантическая ошибка — "Он живет в небольшом городке у Волги.", морфологическая ошибка — "Увидал, что у санитара нет сапогов — велел дать.", галлюцинация — "Люк останавливает удачу от этого.". Пожалуйста, если ты нашел ошибку, опиши ее максимально подробно, чтобы помочь своему ученику.	Перед тобой предложение на русском языке, в котором, возможно, есть ошибка: {input}. Если в предложении есть ошибка, ответь, к какой категории она относится: морфологическая, синтаксическая, семантическая или галлюцинация. Также объясни, в чем именно ошибка, и исправь предложение. Напиши ответ в следующем формате: "категория: [категория ошибки], объяснение: [твое объяснение], исправленное предложение: [правильное предложение]". Если предложение не содержит ошибок, ответь только "в предложении нет ошибок". Пожалуйста, если ты нашел ошибку, опиши ее максимально подробно, чтобы помочь своему ученику. Пожалуйста, будь внимателен, очень важно найти ошибку.

Table 3. Prompts used for YandexGPT and GigaChat.

Regarding Saiga, we have tested this model only on the Label task. We have tried several prompts, including the ones used for YandexGPT and GigaChat, but the model has demonstrated very poor performance with all of them. Thus, we decided to concentrate on evaluating YandexGPT and GigaChat.

3.4. Evaluation scheme

3.4.1 Automated evaluation

Two tasks, namely Label and Category ones, allow us to evaluate the models' performance automatically, since the true labels are provided in the dataset. For these tasks, we have used two metrics: accuracy and Matthew's correlation coefficient (MCC), which is used for unbalanced datasets.

With these metrics, we evaluated 2 of 8 scenarios: label and label+category ones.

3.4.2 Human evaluation

We can not use any metrics to evaluate the quality of the models' explanations and corrections. Hence, the remaining 6 scenarios needed human evaluation, as well as automated one.

To assess the models' explanations and corrections, we annotated 50 examples of each model's answers for each scenario. In each case, this sample of 50 explanations/corrections was chosen from the output of the best (according to the metrics on the label task) run for each scenario.

For evaluating the scenarios with explanations, we used our own labelling scheme. Figure 1 on the next page illustrates the scheme, as well as two examples of evaluation. The maximum score of an explanation is 8. We assess four parts of an explanation, which are localisation of an error, naming of incorrect categories, naming of categories with which the incorrect ones should be replaced and correction. Each of these parts can be labelled 0-2 points, where 1 point is given for the presence of the part, and 1 point is given for the correctness of the part. In some cases, 0.5 points can also be given.

To assess the scenarios with corrections, but without explanations, we used the scheme with 2 scores maximum: one for correction being given/not given, and one for correction being correct/not correct.

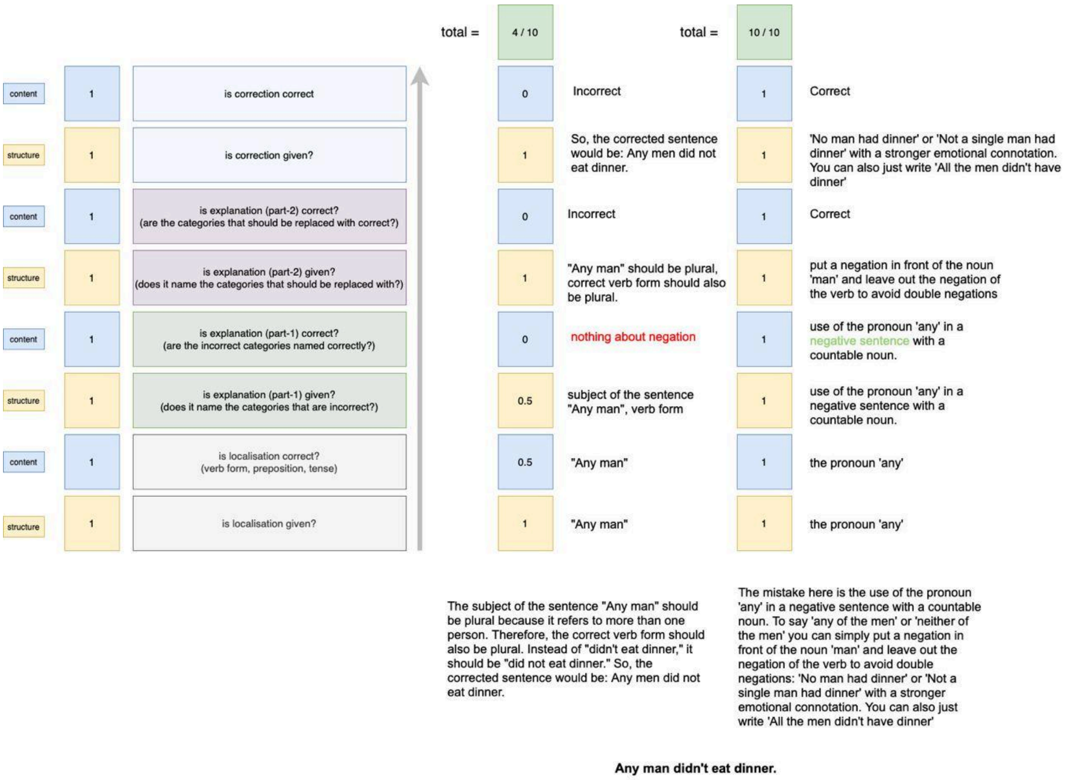
total =
4 / 10
total =
10 / 10
content
1
is correction correct
0
Incorrect
1
Correct
structure
1
is correction given?
1
So, the corrected sentence would be: Any men did not eat dinner.
1
'No man had dinner' or 'Not a single man had dinner' with a stronger emotional connotation. You can also just write 'All the men didn't have dinner'
content
1
is explanation (part-2) correct? (are the categories that should be replaced with correct?)
0
Incorrect
1
Correct
structure
1
is explanation (part-2) given? (does it name the categories that should be replaced with?)
1
"Any man" should be plural, correct verb form should also be plural.
1
put a negation in front of the noun 'man' and leave out the negation of the verb to avoid double negations
content
1
is explanation (part-1) correct? (are the incorrect categories named correctly?)
0
nothing about negation
1
use of the pronoun 'any' in a negative sentence with a countable noun.
structure
1
is explanation (part-1) given? (does it name the categories that are incorrect?)
0.5
subject of the sentence "Any man", verb form
1
use of the pronoun 'any' in a negative sentence with a countable noun.
content
1
is localisation correct? (verb form, preposition, tense)
0.5
"Any man"
1
the pronoun 'any'
structure
1
is localisation given?
1
"Any man"
1
the pronoun 'any'
The subject of the sentence "Any man" should be plural because it refers to more than one person. Therefore, the correct verb form should also be plural. Instead of "didn't eat dinner," it should be "did not eat dinner." So, the corrected sentence would be: Any men did not eat dinner.
The mistake here is the use of the pronoun 'any' in a negative sentence with a countable noun. To say 'any of the men' or 'neither of the men' you can simply put a negation in front of the noun 'man' and leave out the negation of the verb to avoid double negations: 'No man had dinner' or 'Not a single man had dinner' with a stronger emotional connotation. You can also just write 'All the men didn't have dinner'
Any man didn't eat dinner.

Figure 1. The labelling scheme to evaluate the models' explanations.

It is important to state that this thesis is a part of research conducted by 4 students. To ensure that an individual annotation is credible we have measured the agreement between 4 annotators. The obtained values of Krippendorff's alpha were found sufficient to ensure the reliability of individual annotation scores.

4. Results

4.0. Experiments with Saiga

To start with, we wanted to evaluate the grammar knowledge of the Saiga model, not only of YandexGPT and GigaChat. Unfortunately, we stopped after the first scenario (one with only labels to be predicted), as we obtained insufficient results. We ran this scenario with different prompts, and the metrics in the best case were the following (the mean of metrics for 3 runs is given):

Accuracy	0.6297
MCC	0.0269

Table 4. Accuracy and MCC for labels predicted by Saiga.

As these results mean nearly random prediction (MCC is almost 0), we decided to concentrate on other models. Their performance will be reviewed in the next four subsections. In all the figures given, ‘l’ stands for label, ‘cat’ – for category, ‘e’ – for explanation, ‘cor’ – for correction.

4.1 Label

The Label task is for a model to predict the class of a sentence: acceptable or unacceptable. This task is present in all our scenarios, so for evaluating it we have measured accuracy and MCC for predicted labels in case of each scenario. It is important to note that we have run each scenario five times and calculated the mean and the standard deviation of each metric in order to avoid randomness. The metrics for the two models will be presented below.

4.1.1 YandexGPT results

Figure 2 on the next page shows the average metrics obtained for the Label task in the case of YandexGPT. The obtained standard deviations for 5 runs can be found in Appendix B.

In general, both metrics differ not that much across different scenarios. However, it can be seen that YandexGPT predicts labels with the highest accuracy and MCC in the third scenario (Label+explanation). The lowest metrics are observed in the case of the second scenario, Label+category.

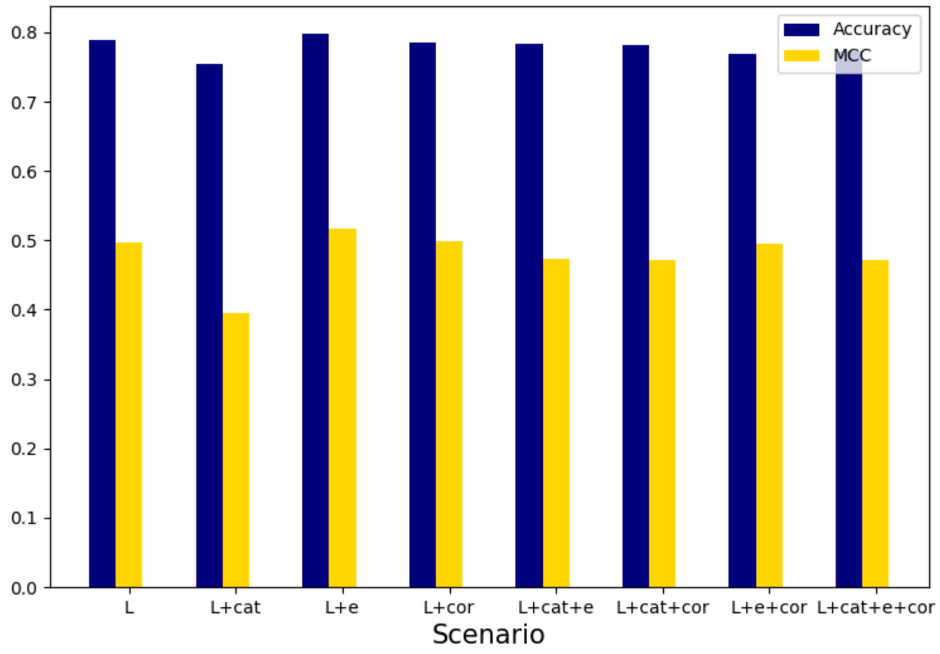

Figure 2. Accuracy and MCC of labels predicted by YandexGPT.

4.1.2 GigaChat results

Figure 3 shows the metrics obtained for the Label task in the case of GigaChat. The obtained standard deviations for 5 runs can be found in Appendix B.

In general, the performance of GigaChat on this task is worse than that of YandexGPT. The difference in metrics across different scenarios is also more significant, it is approximately 0.3 for accuracy and more than 0.2 for MCC. The metrics are the highest in the first (Label) scenario and the lowest in the Label+category+explanation one. It is also observed that the performance on the Label task is generally worse in complex scenarios (those which include 3-4 tasks).

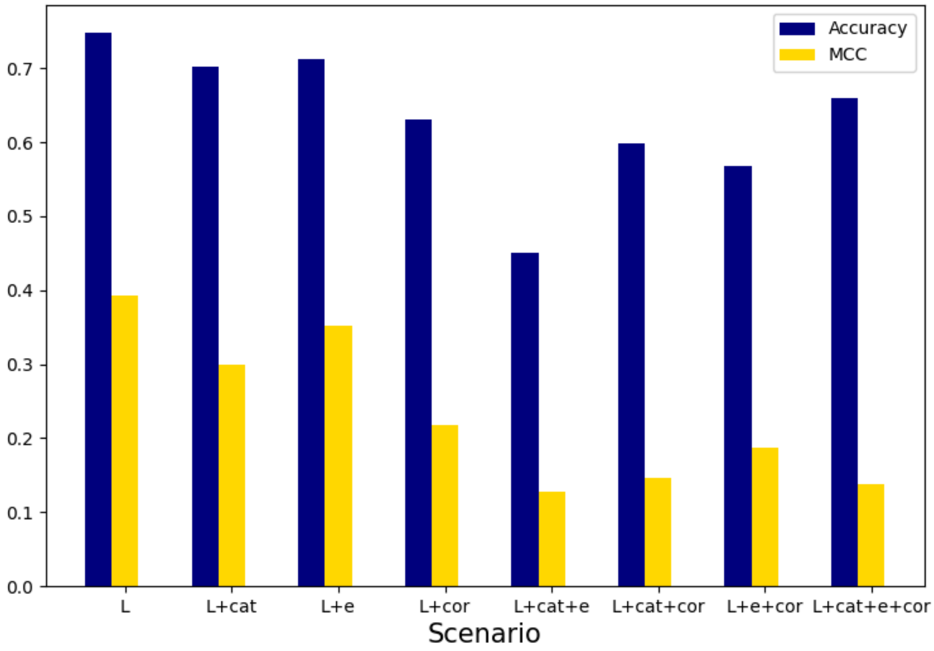

Figure 3. Accuracy and MCC of labels predicted by GigaChat.

4.2 Category

The Category task is the task where we prompt the models to predict the category of an error: Morphology, Syntax, Semantics or Hallucination.

To evaluate the models' performance on the Category tasks, we have measured accuracy of predicted categories in 4 scenarios: Label+category, Label+category +explanation, Label+category+correction, Label+category+explanation+correction. The obtained results for the models are described below.

Figure 4 on the next page illustrates the accuracy of error categories predicted by both GigaChat and YandexGPT. As can be seen, overall YandexGPT has performed better on this task again. In addition to it, the average accuracies vary less in the case of YandexGPT (range is less than 0.1) than in the case of GigaChat (range is more than 0.3).

As can be seen, the highest accuracies were obtained in the same scenario (Label+category) by both models. However, the worst scenario for YandexGPT was the 'Label+category+explanation+correction' one, and for GigaChat this was the

‘Label+category+explanation’ one (interestingly, the accuracy was double lower here than in other scenarios).

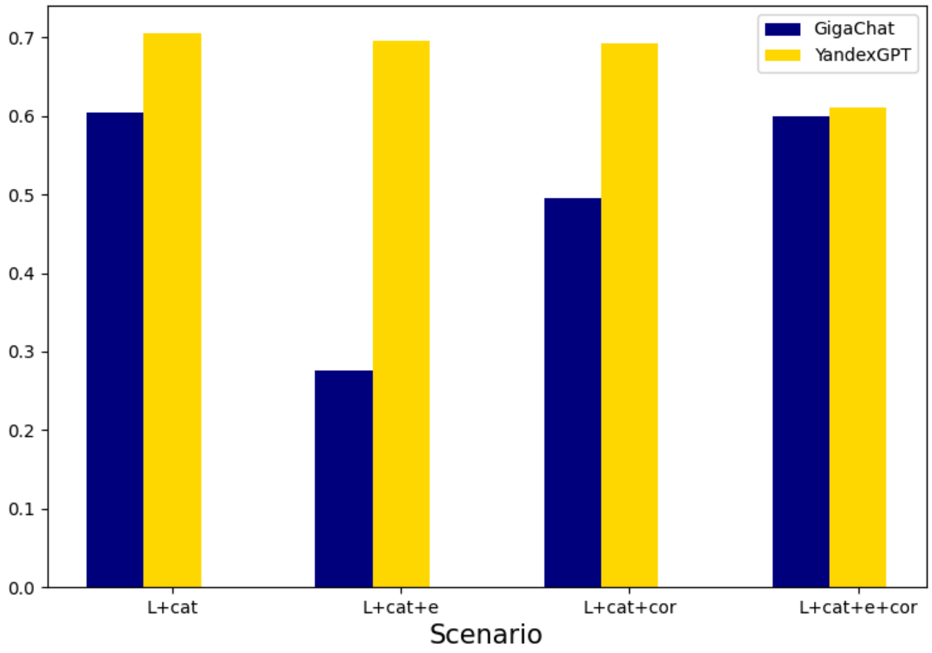

Figure 4. Accuracy of error categories predicted by GigaChat and YandexGPT.

4.3 Explanation

In the explanation task, we prompted models to explain why a sentence is incorrect, in case it is. This task was applied in 4 scenarios in total: ‘label+explanation’, ‘label+category+explanation’, ‘label+explanation+correction’ and ‘label+category+ explanation+correction’.

As it was stated in the Methods and Materials section, we used our own labelling scheme from 0 to 8 points to evaluate the models’ explanations. However, 2 of 8 points are somehow bonus, because they assess the presence and correctness of correction, which is not mandatory while giving an explanation of error. So, below I will present two scores for each scenario for each model: one where maximum is 6 and one where maximum is 8.

First, we will look at the results of YandexGPT. Next, the Gigachat’s results will be reported.

4.3.1 YandexGPT results

Figure 5 below illustrates the average scores of 50 annotated explanations for each of the four scenarios. All the scores are also listed in the Appendix B.

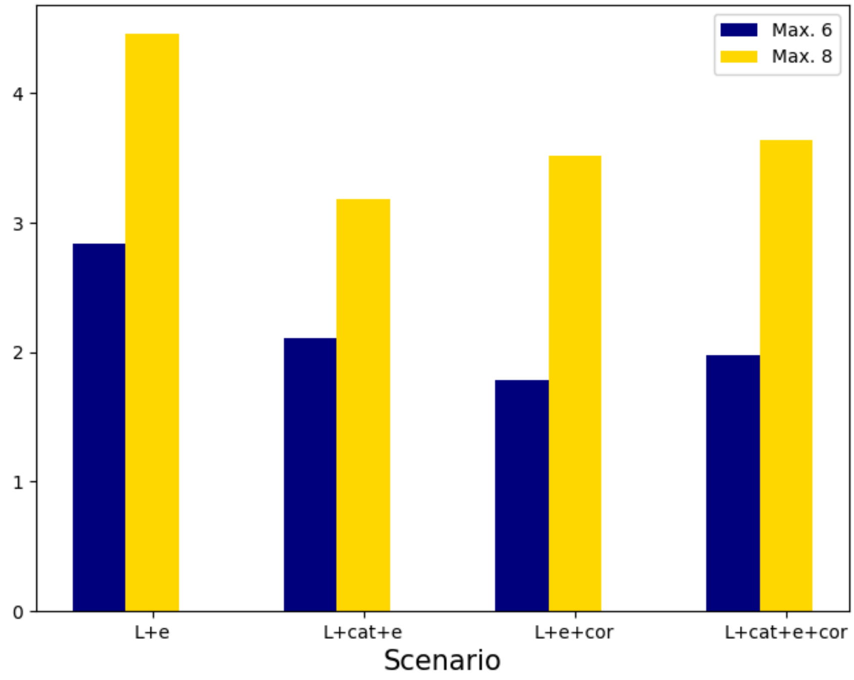

Figure 5. Average scores of explanations given by YandexGPT.

It can be seen that YandexGPT's explanations are not always sufficient. For example, 4/8 points can be scored if a model incorrectly gives every part of explanation (it wrongly localises the error, wrongly states which categories are incorrect, wrongly names the categories for replacement and gives wrong correction).

The highest score of explanations can be observed in the scenario where the model is prompted only for them. The scenario with the lowest score differs for the two scores: the lowest 'Max. 6' score is observed in the 'label+explanation+correction' scenario, while the lowest 'Max. 8' score is in the 'label+category+explanation' scenario. It can be explained by the fact that the third and the fourth scenarios explicitly ask the model for a correction.

The decline in scores in the last two scenarios is rather expectable because in these scenarios YandexGPT tends to give shorter explanations. For example, here are the typical explanations given by the model in 4 scenarios:

(1) *Label+explanation: В этом предложении есть ошибка. Ошибка содержится в слове «приобретал». Это слово не согласуется с остальными по времени. Чтобы исправить ошибку, нужно поставить глагол в прошедшее время, единственное число и согласовать его с местоимением «он»: «Когда после окончания школы мой друг поступил на завод, он за короткое время **приобрел** квалификацию токаря».*

(2) *Label+category+explanation: В предложении есть ошибка, относящаяся к категории «морфология». Слово «хоп» — это звукоподражание, оно не склоняется. В данном случае ошибочное употребление формы родительного падежа «хопа» приводит к нарушению согласования слов в предложении. **Верный вариант:** «Если вы можете видеть их, хоп не должен быть таким плохим».*

(3) *Label+explanation+correction: ошибка в употреблении причастия прошедшего времени, которое не согласовано с определяемым словом в роде и числе.*

(4) *Label+category+explanation+correction: Объяснение: ошибка в структуре предложения, неправильное употребление союза «чтобы».*

4.3.2 GigaChat results

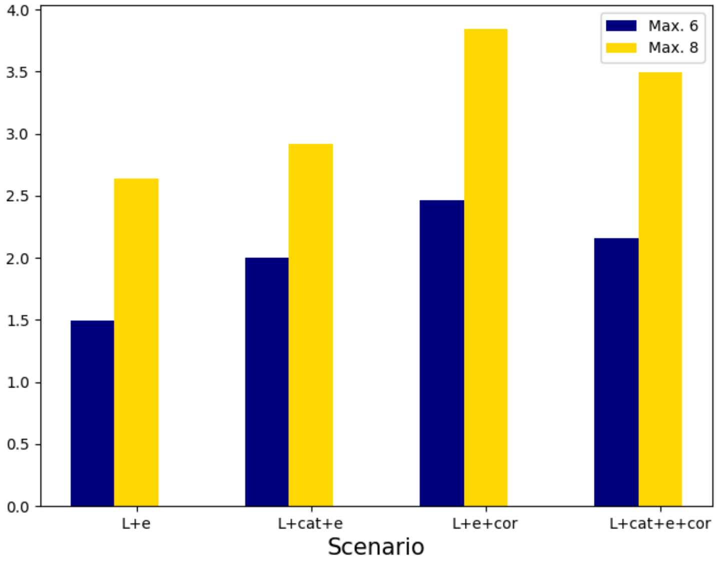

Figure 6. Average scores of explanations given by GigaChat.

Figure 6 above shows the average scores of 50 annotated explanations for each of the four scenarios. All the scores are also listed in the Appendix B.

It is apparent that the quality of GigaChat's explanations is even worse than in the case of YandexGPT. Even the scores with 2 points for corrections are lower than 4.0 in every scenario.

The highest score of explanations (in terms of both scores) can be observed in the scenario where the model is prompted for them as well as for corrections. Interestingly, the scenario with the lowest score is also the same for the two scores: this is the 'label+explanation' one.

4.4 Correction

In the correction task, the models were prompted to correct the error in a sentence, if it is incorrect. This task was applied in 4 scenarios in total: 'label+correction', 'label+category+correction', 'label+explanation+correction' and 'label+category+ explanation+correction'.

As it was stated in the Methods and Materials section, we used a scheme with 0-2 scores for assessing the quality of corrections given. Figure 7 depicts the average score of 50 annotated corrections for each of the four scenarios for both models evaluated.

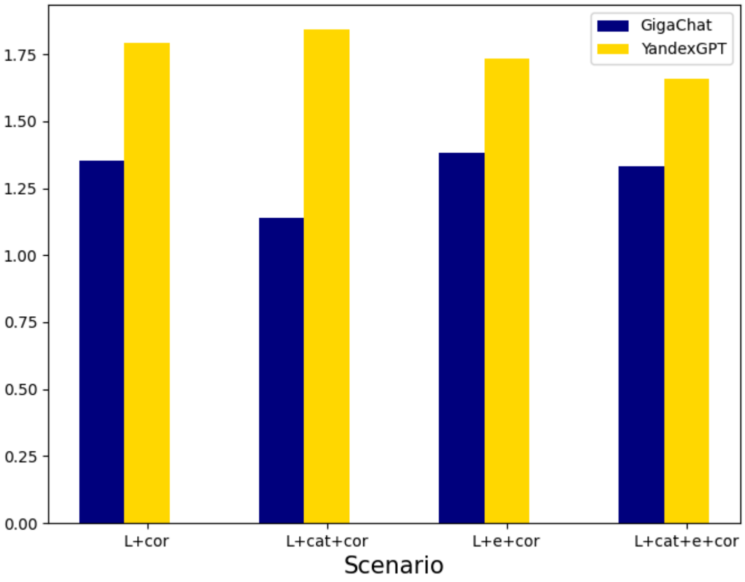

Figure 7. Average score of corrections given by YandexGPT and GigaChat.

Again, YandexGPT performs better than GigaChat. Both models almost always produce a sentence, which is supposed to be the correction of the initial one. However, this sequence may not be the correct version of the sentence given, and it is especially noticeable in the case of Gigachat, where average scores are lower than 1.4/2 in all four scenarios. For example, GigaChat, being prompted with the sentence *Рассмотрим два высказывания, которых доказать эквивалентность не составит труда.*, gave the following ‘correction’: *Рассмотрим два высказывания, которые доказать эквивалентность не составит труда.*

Gigachat gives the most decent corrections in two scenarios: one which requires only this and a label (not prediction of error category or explanation of an error as well) and one which requires a label with both explanation and correction.

On the other hand, the average score of YandexGPT’s corrections is slightly higher in the more complex scenario, where the model is also prompted to predict the category of an error. However, its performance in the ‘label+correction’ scenario is also high.

The worst scenario for GigaChat is the ‘label+category+correction’ one. YandexGPT gives the lowest scored corrections when being prompted with the most complex scenario, which requires solving all the four tasks.

5. Discussion

In this section, I will discuss the obtained results in order to give some recommendations about how the considered models could be effectively used for different grammar tasks.

First of all, it can be seen that different models can be useful for grammar tasks to varying degrees. Whereas YandexGPT achieves rather high results on all the tasks except explanations, GigaChat performs less successfully. So, we can not state that all Russian LLMs can be helpful while learning or teaching grammar.

Next, the models cope with different tasks with varying degrees of success, too. While both evaluated LLMs generally handle the label prediction task adequately, the performance on other tasks (category of an error prediction, errors explanation and correction of them) is not always convincing. Here, we will discuss the performance of the LLMs regarding each task and suggest which scenarios are more beneficial in each case.

5.1 The Label task

Here, we can not give the same recommendations for different models, because two evaluated LLMs behave differently. If we want YandexGPT to predict whether a sentence is correct or incorrect, it is most beneficial to prompt the model not only to classify a sentence, but to provide an explanation of an error as well. On the contrary, in the case of GigaChat it is more useful to ask the model only to answer whether a sentence is correct or incorrect.

5.2 The Category task

For predicting the category of a particular error, the recommendation is the same for both models. As our results show, it is most effective to only prompt the model to predict if a sentence is correct or incorrect and give an error category in the latter case. Giving a model other tasks (explanation and correction) does not contribute to a better performance on the error category prediction.

5.3 The Explanation task

In the case of eliciting error explanations, the recommendations also differ for the models. With YandexGPT, it is recommended to prompt the model to only classify a sentence as (in)correct and provide an explanation of an error, if it is present in the sentence. On the other hand, to obtain the best explanations from GigaChat, it is advisable to ask the model not only about a label and an explanation itself, but about an error correction as well.

5.4 The Correction task

Finally, in the case of correcting errors in sentences, the recommendations are not similar for the two models also. For YandexGPT, it is evident that the most effective way is to prompt the model not only for a label and an error correction, but for a category of an error as well. To get the best corrections from GigaChat, however, it is most beneficial to prompt the model for a label, an explanation and a correction simultaneously.

Overall, for all the four grammar tasks it is recommended to use YandexGPT, as its performance is more impressive. However, this model is still not suitable for all the tasks, because the quality of explanations given by the model is probably not sufficient.

6. Conclusion

This thesis evaluates whether large language models designed for the Russian language possess sufficient grammar knowledge to be used as assistants while teaching grammar of Russian. We wanted to answer the questions, whether LLMs are capable of judging sentences as (un)acceptable and explain their decisions. We also aimed to provide some recommendations about which scenarios can be more beneficial for the models' answers quality in terms of different grammar tasks: acceptability judgments, prediction of error categories, explanation of errors and their correction.

We have tested two models, YandexGPT and GigaChat. Initially, we have also tried to test the third model which was Saiga. We have used prompting to get the answers from these models. During experiments, we prompted the LLMs with several combinations of the 4 tasks listed above and evaluated the models' performance on these scenarios using accuracy, Matthew's correlation coefficient and human annotation based on our labelling scheme.

As our results show, the LLMs demonstrate different levels of performance, with YandexGPT being more competent than GigaChat. Furthermore, both models do not cope with different scenarios equally. The easiest task for the models is to predict whether a given sentence is correct or incorrect, while providing explanations of errors is the most difficult one.

Regarding recommendations on how to apply LLMs for grammar tasks, they differ for the two models studied and were described in the Discussion section in detail. Overall, our results show that large language models can be applied in the field of grammar teaching or correction with caution, since they do not always demonstrate sufficient knowledge. These models can be used successfully for labelling a sentence as acceptable or unacceptable, correcting the errors in it or naming categories of them (only YandexGPT is suitable for the latter two), but probably not for giving explanations of errors.

As a practical outcome of this study, we also present our labelling scheme, which can be used for evaluation of error explanations given by language models.

References

Bang, Y., Cahyawijaya, S., Lee, N., Dai, W., Su, D., Wilie, B., ... & Fung, P. (2023). A multitask, multilingual, multimodal evaluation of chatgpt on reasoning, hallucination, and interactivity. arXiv preprint arXiv:2302.04023.

Bermel, N., & Knittl, L. (2012). Corpus frequency and acceptability judgments: A study of morphosyntactic variants in Czech. Corpus Linguistics and Linguistic Theory, 8(2), 241-275.

Chomsky, N. (2002). *Syntactic structures*. Mouton de Gruyter.

Dentella, V., Murphy, E., Marcus, G., & Leivada, E. (2023). Testing AI performance on less frequent aspects of language reveals insensitivity to underlying meaning. arXiv preprint arXiv:2302.12313.

Hendy, A., Abdelrehim, M., Sharaf, A., Raunak, V., Gabr, M., Matsushita, H., ... & Awadalla, H. H. (2023). How good are gpt models at machine translation? a comprehensive evaluation. arXiv preprint arXiv:2302.09210.

Hu, H., Zhang, Z., Huang, W., Lai, J. Y. K., Li, A., Ma, Y., ... & Wang, R. (2023). Revisiting Acceptability Judgements. arXiv preprint arXiv:2305.14091.

Kong, A., Zhao, S., Chen, H., Li, Q., Qin, Y., Sun, R., & Zhou, X. (2023). Better zero-shot reasoning with role-play prompting. arXiv preprint arXiv:2308.07702.

Kocoń, J., Cichecki, I., Kaszyca, O., Kochanek, M., Szydło, D., Baran, J., ... & Kazienko, P. (2023). ChatGPT: Jack of all trades, master of none. Information Fusion, 99, 101861.

Liu, P., Yuan, W., Fu, J., Jiang, Z., Hayashi, H., & Neubig, G. (2023). Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. ACM Computing Surveys, 55(9), 1-35.

Mikhailov, V., Shamardina, T., Ryabinin, M., Pestova, A., Smurov, I., & Artemova, E. (2022). RuCoLA: Russian corpus of linguistic acceptability. arXiv preprint arXiv:2210.12814.

Ortega-Martín, M., García-Sierra, Ó., Ardoiz, A., Álvarez, J., Armenteros, J. C., & Alonso, A. (2023). Linguistic ambiguity analysis in chatgpt. arXiv preprint arXiv:2302.06426.

Radford, A., Narasimhan, K., Salimans, T., & Sutskever, I. (2018). Improving language understanding by generative pre-training.

Santu, S. K. K., & Feng, D. (2023). Teler: A general taxonomy of llm prompts for benchmarking complex tasks. arXiv preprint arXiv:2305.11430.

Schütze, C. (2016). *The empirical base of linguistics: Grammaticality judgments and linguistic methodology*. Language Science Press.

Song, Y., Krishna, K., Bhatt, R., Gimpel, K., & Iyyer, M. (2023). Gee! grammar error explanation with large language models. arXiv preprint arXiv:2311.09517.

Sprouse, J. (2018). Acceptability judgments and grammaticality, prospects and challenges. Syntactic structures after, 60, 195-224.

Trotta, D., Guarasci, R., Leonardelli, E., & Tonelli, S. (2021). Monolingual and cross-lingual acceptability judgments with the Italian CoLA corpus. arXiv preprint arXiv:2109.12053.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention is all you need. Advances in neural information processing systems, 30.

De Villiers, P. A., & de Villiers, J. G. (1972). Early judgments of semantic and syntactic acceptability by children. Journal of Psycholinguistic Research, 1(4), 299-310.

Volodina, E., Mohammed, Y. A., & Klezl, J. (2021). DaLAJ-a dataset for linguistic acceptability judgments for Swedish: Format, baseline, sharing. arXiv preprint arXiv:2105.06681.

Warstadt, A., Singh, A., & Bowman, S. R. (2019). Neural network acceptability judgments. Transactions of the Association for Computational Linguistics, 7, 625-641.

Warstadt, A., Parrish, A., Liu, H., Mohananey, A., Peng, W., Wang, S. F., & Bowman, S. R. (2020). BLiMP: The benchmark of linguistic minimal pairs for English. Transactions of the Association for Computational Linguistics, 8, 377-392.

Zhang, Z., Liu, Y., Huang, W., Mao, J., Wang, R., & Hu, H. (2023). MELA: Multilingual Evaluation of Linguistic Acceptability. arXiv preprint arXiv:2311.09033.

Zhong, Q., Ding, L., Liu, J., Du, B., & Tao, D. (2023). Can chatgpt understand too? a comparative study on chatgpt and fine-tuned bert. arXiv preprint arXiv:2302.10198.

RussianNLP/rucola · Datasets at Hugging Face. (2023, August 4). https://huggingface.co/datasets/RussianNLP/rucola

IlyaGusev/saiga_7b_lora · Hugging Face. (2023, September 4). https://huggingface.co/IlyaGusev/saiga_7b_lora

YandexCloud. (n.d.) https://yandex.cloud/en/

Studio – создать чатбот или получить доступ к голосовому API. (n.d.). https://developers.sber.ru/studio/

Appendix

A Source code and data

All the code used for prompting models, as well as some files with annotation of the models' explanations and corrections can be found in the following github repository: https://github.com/marianetta/LLMs_and_grammar

B Obtained metrics

1. *Label*

	Accuracy	MCC
Label	0.7897 ∓ 0.0012	0.497 ∓ 0.0032
Label+category	0.7547 ∓ 0.0187	0.3958 ∓ 0.0567
Label+explanation	0.798 ∓ 0.0178	0.5172 ∓ 0.0382
Label+correction	0.7856 ∓ 0.0213	0.4996 ∓ 0.0649
Label+category+explanation	0.7832 ∓ 0.0199	0.474 ∓ 0.0544
Label+category+correction	0.7813 ∓ 0.0073	0.4707 ∓ 0.0203
Label+explanation+correction	0.7685 ∓ 0.0049	0.4945 ∓ 0.0101
Label+category+explanation+correction	0.7725 ∓ 0.0045	0.4713 ∓ 0.0087

Table 1. Mean and std of accuracy and MCC of labels predicted by YandexGPT

	Accuracy	MCC
Label	0.7482 ∓ 0.0059	0.3923 ∓ 0.0354
Label+category	0.7028 ∓ 0.0042	0.2991 ∓ 0.0111
Label+explanation	0.7122 ∓ 0.003	0.3516 ∓ 0.0042
Label+correction	0.6317 ∓ 0.0471	0.2179 ∓ 0.0296

Label+category+explanation	0.4502 ∓ 0.0109	0.1282 ∓ 0.0892
Label+category+correction	0.5988 ∓ 0.0372	0.1463 ∓ 0.0469
Label+explanation+correction	0.5679 ∓ 0.069	0.187 ∓ 0.0325
Label+category+explanation+correction	0.659 ∓ 0.0132	0.1373 ∓ 0.0934

Table 2. Mean and std of accuracy and MCC of labels predicted by GigaChat.

2. *Category*

Scenario	Accuracy
Label+category	0.7056 ∓ 0.0027
Label+category+explanation	0.6959 ∓ 0.0312
Label+category+correction	0.6925 ∓ 0.0272
Label+category+explanation+correction	0.6117 ∓ 0.0529

Table 3. Mean and std of accuracy of categories predicted by YandexGPT.

Scenario	Accuracy
Label+category	0.605 ∓ 0.0026
Label+category+explanation	0.2758 ∓ 0.0123
Label+category+correction	0.4948 ∓ 0.0403
Label+category+explanation+correction	0.5993 ∓ 0.0208

Table 4. Mean and std of accuracy of categories predicted by GigaChat.

3. *Explanation*

Scenario	Score (out of 6)	Score (out of 8)
Label+explanation	2.8431	4.4608
Label+category+explanation	2.1078	3.1863
Label+explanation+correction	1.7843	3.5196
Label+category+explanation+correction	1.9804	3.6373

Table 5. Average scores of explanations given by YandexGPT.

Scenario	Score (out of 6)	Score (out of 8)
Label+explanation	1.4902	2.6373
Label+category+explanation	2	2.9216
Label+explanation+correction	2.4608	3.8431
Label+category+explanation+correction	2.1569	3.4902

Table 6. Average scores of explanations given by GigaChat.

4. Correction

Scenario	Score (out of 2)
Label+correction	1.7941
Label+category+correction	1.8431
Label+explanation+correction	1.7353
Label+category+explanation+correction	1.6569

Table 7. Average scores of corrections given by YandexGPT.

Scenario	Score (out of 2)
Label+correction	1.353
Label+category+correction	1.1373
Label+explanation+correction	1.3824
Label+category+explanation+correction	1.3333

Table 8. Average scores of corrections given by GigaChat.