Bulgarian Grammar Error Correction with Data Augmentation and Machine Translation Techniques

Bozhidar Klouchek
The University of Manchester
klouchek.bozhidar@gmail.com

Riza Batista-Navarro
The University of Manchester
riza.batista@manchester.ac.uk

Abstract

Grammar Error Correction (GEC) in Bulgarian is particularly difficult because of the lack of specialised linguistic resources and the highly inflected nature of the language. To facilitate research in the field we release three datasets created using data augmentation techniques inspired from phonetic and syntactic phenomena in the language. The datasets include a comprehensive machine-readable dictionary and two error correction datasets containing examples of spelling and grammar mistakes, respectively. Additionally, we employed an encoder-decoder transformer architecture, specifically multilingual T5 (mT5), to address the task of GEC using Neural Machine Translation (NMT). The proposed fine-tuned model outperforms contemporary large language model (LLM)-based solutions such as GPT and BgGPT by scoring an F0.5-score of 68.18%. It is also the most preferable in terms of output readability and correctness according to the human-based evaluation we conducted.

1 Introduction

Bulgarian is a highly inflected language, i.e., words take on different forms to convey information relating to gender, number, article, tense, case and other properties. Because of this, a slight syntactic difference can drastically change a word's meaning. For example, the word 'approach' in Bulgarian ('доближавам') has 51 different forms (Simov et al., 2004), all carrying different nuances about the speaker, the action's time frame and the level of emotion used in uttering the word. This, along with Bulgarian's intricate grammar, spelling and punctuation, makes mastering the language a unique challenge.

Despite the fact that as of 2011, the literacy rate in Bulgaria is 98.7% (National Statistical Institute, 2023), the language literacy performance of young Bulgarian students learning the language is lower than the average for tested countries (OECD, 2023), placing Bulgaria's mean score significantly below the average. This is troubling, as literacy is the foundation of language proficiency and is crucial for employability (Suarta et al., 2017), education (Castro et al., 2011) and social integration (Abdul-Rahaman et al., 2023).

Bulgarian Natural Language Processing (NLP) researchers could help alleviate this problem by creating: (a) linguistic resources, i.e., datasets, that facilitate the development of language literacy tools, and (b) error correction tools for Bulgarian. However, such resources and tools are currently lacking.

Datasets that could potentially facilitate the development of Bulgarian NLP tools include machine-readable dictionaries and error correction data. An official Bulgarian dictionary called the Institute for Bulgarian Language Online Dictionary[1] has been made publicly available. However, it comes with shortcomings that make it unsuitable for supporting the development of NLP tools. These include its inability to recognise words unless they are in their base word form, low confidence in recognising words that do not have a Bulgarian etymological origin, and lack of access to the entire word collection (preventing download by bulk). Meanwhile, error correction data is often required by systems that are developed or trained to assess language validity. This pertains to datasets that contain examples of spelling and grammar mistakes. There are many datasets of this kind for high-resource languages, e.g. English (Geertzen et al., 2013; Granger et al., 2009) and German (Meisel, 2020). However, no such Bulgarian resource is available.

Error correction tools are underpinned by models that verify a text's linguistic validity, focussing on different aspects of the language, including punctuation, spelling and grammar. Bulgarian spelling correction has been explored using noisy text mod-

[1] https://ibl.bas.bg/rbe/lang/en/

els (Gerdjikov et al., 2013). This approach was developed using the IMPACT BG dataset[2] which consists of 19^{th} century Bulgarian newspaper articles, which are not indicative of modern Bulgarian communication. Grammar Error Correction (GEC) research for Bulgarian is scarce, most likely because of the short supply of error correction data and the inflectional nature of the language.

In this paper we aim to address the research gap caused by the lack of appropriate linguistic resources and absence of established solutions for error correction by releasing open-source datasets supporting language literacy and training a generative model for automatic GEC. We hope that this will encourage other members of the research community to build and compare solutions for Bulgarian language literacy tasks.

Our contributions include: (1) the creation of three datasets including a machine-readable Bulgarian dictionary that includes word inflections and Part-of-Speech tags, and two datasets produced using data augmentation, both of which contain pairs of erroneous and corrected sentences, one with spelling errors and the other with grammatical mistakes; and (2) the development and evaluation of a GEC solution based on fine-tuning a multilingual T5 (mT5) model (Xue et al., 2021) for neural machine translation of erroneous to correct text. The model,[3] along with the dictionary,[4] spelling error[5] and grammar error[6] datasets, are all open-source and available for public use.

2 Related Work

Dataset creation and error correction techniques are both pivotal for the success of automated language literacy tools. We review how these have been approached for Bulgarian and similar highly inflected languages.

2.1 Bulgarian Datasets

Below, we provide an overview of previously reported work on the development of Bulgarian linguistic resources and error correction datasets.

Linguistic Resources. Open-source Bulgarian linguistic resources have been published from as early as two decades ago. Among these is the BulTreeBank project (Simov et al., 2002), developed by the Bulgarian Academy of Sciences (BAS), which is considered to be the first successful initiative for large-scale curation for linguistic resources in Bulgarian NLP. The CLaRK system (Simov et al., 2003) is another notable achievement, presenting an automated system for corpora development that includes part-of-speech (POS) tagging and dependency extraction, utilising regular cascaded grammars. The CLaRK system is particularly useful for high-quality POS tagging in Bulgarian; we employed it in our work to identify candidates that can form the basis for inducing artificial errors.

More recent projects developed textual corpora that were drawn from specific domains such as law and medicine (Koeva et al., 2020; Boytcheva et al., 2020), as well as speech corpora (Dimitrova, 2021). However, no established Bulgarian error correction datasets have been released, hindering the progress of NLP researchers on error correction tasks.

Error Correction Datasets. Error correction data is particularly difficult to come by, as it necessitates a sophisticated approach to collection and/or generation of erroneous use of language. Systems using high-resource languages, like English (Dolgova and Mueller, 2019) and Chinese (Rao et al., 2018) rely on authentic *learner data* created by learners of the language, which can be then annotated manually. Low-resource language systems, however, tend to use *synthetic data* generated through *data augmentation*. This technique does not require language learners, rather, it generates the error correction data automatically by either:

- round-trip translation from error-free text, resulting in ungrammatical sentences (Lichtarge et al., 2019), or
- directly inducing errors in error-free text (Grundkiewicz and Junczys-Dowmunt, 2019; Lee and Seneff, 2008; Izumi et al., 2003).

Data augmentation was proven to be especially useful for low-resource languages (Solyman et al., 2023), as it provides a sustainable solution to the data scarcity problem. In this project, we chose to create artificial erroneous data by directly inducing grammatical errors based on predetermined linguistic rules, as there is a distinct lack of publicly

[2] https://www.digitisation.eu/datasets/impact-language-resources/

[3] https://huggingface.co/thebogko/mt5-finetuned-bulgarian-grammar-mistakes

[4] https://huggingface.co/datasets/thebogko/bulgarian-dictionary-2024

[5] https://huggingface.co/datasets/thebogko/bulgarian-spelling-mistakes

[6] https://huggingface.co/datasets/thebogko/bulgarian-grammar-mistakes

accessible learner data from humans.

2.2 Grammar Error Correction (GEC)

GEC approaches focus on transforming erroneous text to its correct version by identifying mistakes and recommending suggestions. These are typically based on machine translation (MT) methods.

Machine Translation Models. Statistical Machine Translation (SMT) is a probabilistic approach applied to GEC which, given an erroneous text sequence $e_1, e_2, \dots e_m$, identifies the corrected text sequence $c_1, c_2, \dots c_n$ that maximises the probability $p(c_1, c_2, \dots c_n | e_1, e_2, \dots e_m)$. This approach is often supported by a language model (Wang et al., 2021), ensuring that the corrections are fluent. The first error correction work based on SMT focussed on noun errors (Brockett et al., 2006).

Neural Machine Translation (NMT) differs from SMT in that it utilises *neural networks* to generate corrected text output (target) given erroneous input (source). Its strength lies in the ability of neural networks to generalise, allowing NMT systems to perform much better than SMT in correcting unseen error types (Wang et al., 2021). The first time it was used for GEC (Yuan and Briscoe, 2016) was ten years after SMT was first attempted, becoming the predominant approach to solving the error correction task. Diverse architectures have been used in NMT, such as recurrent neural networks (RNNs) (Yuan and Briscoe, 2016), convolutional neural networks (CNNs) (Chollampatt and Ng, 2018; Solyman et al., 2019) and transformers (Zhao et al., 2019; Grundkiewicz et al., 2019). Because of the success of NMT approaches we utilise it for our GEC task.

GEC for Highly Inflected Languages. GEC research for Bulgarian is scarce, likely because of the short supply of error correction data and the inflectional nature of the language. Some efforts have been made to detect noun-adjective disagreement (Borisova et al., 2014) and to investigate how grammars can be used for error correction (Kubon and Plátek, 1994), but at the time of writing no machine translation approach has been proposed for GEC in Bulgarian.

Our work aims to remedy this by training a transformer-based model on a large collection of grammar error mistakes and their corresponding corrections, motivated by studies showing that MT-based error correction systems for morphologically rich languages require large amounts of training data (Rozovskaya and Roth, 2019). We employed the mT5 model, given that it obtained encouraging results in the correction of highly inflected languages like Ukrainian (Lytvyn et al., 2023).

3 Creating Language Literacy Datasets

One of our objectives is the development of high-quality and open-source datasets that can be used for a diverse range of tasks that support language literacy. We showcase a comprehensive dictionary and two error correction datasets consisting of sentence pairs: one dataset contains spelling errors and the other contains grammatical mistakes.

3.1 A Machine-Readable Dictionary

Our Bulgarian dictionary contains 1,147,600 entries, each with a term and a corresponding part-of-speech (POS) tag. In this project's context, a *term* is defined as either the base form (lemma) of a word or an inflected form; in both cases, we only include single-word terms. This would allow a spell-checking system to perform a simple check for each token from a user's input to determine its validity.

Data Collection and Preprocessing. We firstly collected entries from two major open-source collections[7,8] due to their popularity and sufficient word coverage. It is worth noting that some inflected Bulgarian words, particularly verbs, can be supported by particles. For example, the word 'ям' ('eat') can change to 'ял' in some forms depending on the tense.

- 'щях да съм ял'
- 'бил съм ял'
- 'ял си'

Since the aforementioned dictionaries include these multi-word terms, while a spell-checking system would be expected to judge single tokens' validity on their own, we break up these multi-word terms and only look at unique sequences of characters. In this way we significantly decrease the number of terms, while still maintaining the dictionary's ability to determine if a word is spelled correctly.

Labelling. Additionally, the POS tag of each term is carried over from the sources we used. The tagging scheme includes 11 tags and is based on BulTreeBank's tagging scheme (Simov et al.,

[7] https://slovored.com
[8] https://rechnik.chitanka.info

2004), but was simplified by including only a single capitalised letter for the high-level role of the tag (e.g., `Amsf`, `Ansd`, etc. all conflate to `A` for 'adjective'). Including these tags allows the dictionary to differentiate between homonyms. For example, the word 'син' describes both the adjective 'blue' and the noun 'son', so both are included in the dictionary with different POS tags.

3.2 Error Correction Datasets

Because of the scarcity of learner data in Bulgarian, we propose to collect Bulgarian text data and automatically induce spelling and grammar mistakes using data augmentation techniques. This approach allows us to generate pairs of correct-erroneous sentences, which will be necessary for training models to correct mistakes.

3.2.1 Error-inducement Algorithm

Not all errors can appear in all sentences, as they have specific phonetic, grammatical or lexical requirements. We defined an algorithm for inducing errors that takes a collection of source correct sentences C and a collection of error types T and returns a collection of unique tuples P, each tuple including three elements: a correct sentence $c \in C$, an erroneous sentence e and an error type $t \in T$.

3.2.2 Dataset for Spelling Error Correction

Our spelling error dataset consists of 23,719 pairs of Bulgarian sentences. In each pair, one sentence is the original sentence collected from the source corpus, which is presumed to be correct. The second one is an erroneous version of the correct one, including 1-3 spelling errors of the same type. The dataset spans 7 different error classes based on different linguistic phenomena in Bulgarian and each pair is labelled with one of those classes. To produce this dataset, the steps described below were carried out.

Data Collection and Preprocessing. The source data used to generate this dataset is Bulgarian Wikipedia articles, as we consider the quality of text in Wikipedia as being sufficient for our purposes. Overall, 28 Wikipedia articles were collected. The articles were fed into an spaCy implementation of a preprocessing pipeline specifically for Bulgarian text (Berbatova and Ivanov, 2023). Specifically, the articles underwent sentence segmentation, tokenisation and POS tagging.

To remove noisy sentences, two filters are applied, removing any sentences with fewer than three words or those without any verbs. This eliminated any sentences which are too short to be useful erroneous candidates. A total of 5817 sentences were retained after this step.

Labelling. The seven error types listed below were automatically induced. For incorporating certain types of errors in a sentence, specific sounds or characters need to be present.

1. **Vowel Stress Change.** If a vowel is not in stressed position,[9] change it to the respective vowel counterpart[10] (e.g. 'кръгъл' → 'кръгал').
2. **Assimilation.** If two neighbouring consonants differ in their voice quality,[11] change the former consonant so it follows the voice quality of the latter (e.g. 'постановка' → 'постафка').
3. **Word-final Devoicing.** If there is a voiced consonant at the end of the word, change the consonant to its voiceless form (e.g. 'масив' → 'масиф').
4. **Double Consonant Loss.** If there is a double 'т' or double 'н', remove one of them (e.g. 'полетта' → 'преолета').
5. **Consonant Clusters.** If a specified consonant cluster is present (e.g. 'стн', 'здн', 'щт'), remove 'т' or 'д' (e.g. 'местно' → 'месно').
6. **Random Character.** Introduce a random character into a word (e.g. 'момиче' → 'момгче').
7. **Semantic Change.** If a character replacement, removal, addition or swap operation causes a word to result in a different word, which is spelled correctly, change it (e.g. 'което' ('which') → 'котето' ('kitten')).

It is worth noting that the resulting spelling correction dataset was not used for training any of the models presented in this work. Nevertheless, such a dataset is still necessary for quantifiable evaluation of any spelling correction model and it may prove useful to other members of the research community.

The data distribution presented in Figure 1 shows the frequencies of the different types of spelling errors within the dataset.

[9] A vowel in stressed position is pronounced longer and louder than an unstressed one.

[10] Bulgarian vowels are paired in terms of where they are articulated in the mouth, e.g. 'а' and 'ъ'.

[11] Consonants in Bulgarian are separated into voiced and voiceless, with the majority of them forming pairs.

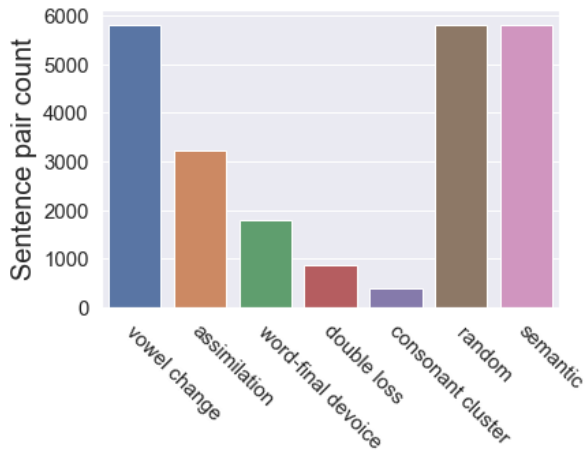

Figure 1: Error type distribution in the spelling error dataset.

3.2.3 Data set for Grammar Error Correction

Our grammar error dataset consists of 7588 error pairs. Similarly to the spelling error dataset, in each pair, the first sentence is the original correct sentence and the second one contains an induced error. Each erroneous sentence has only one induced grammatical error, which can be one of four error types.

Data Collection and Preprocessing. The source of the sentences is a combination of Wikipedia articles and Bulgarian data from the OSCAR project.[12] The same 28 Wikipedia articles in the spelling error dataset were used, in addition to 959,112 documents from the OSCAR dataset. The OSCAR documents were collected from open-source online materials, so the grammatical validity of the data may not be as good as Wikipedia. However, the Wikipedia articles are too similar in terms of writing style. Because of this, the errors induced from them are too similar; thus, including OSCAR diversified the dataset, allowing for a more balanced grammar error distribution.

Preprocessing steps that are similar to those applied on the spelling error dataset were used. Specifically, all documents were fed into a pipeline to perform sentence segmentation, tokenisation and POS tagging. Only sentences with three words or more and contained a token tagged as a verb were kept. Here, the POS tagging component of the CLaRK System (Simov et al., 2003) was used,[13] available via Mate Tools, a toolkit developed by University of Stuttgart's Institute for Natural Language Processing. [14]

	Part-of-Speech (POS) Tag Groups
1	{Ncmsf, Ncmsh}
2	{Pie-os-m, Pie-as-m}, {Pre-os-m, Pre-as-m}, {Prp–s-m, Prp–s-f, Prp–s-n, Prp–p}
3	{V-1p}
4	{Amsi, Afsi, Ansi, A-pi}, {Amsh, Afsd, Ansd, A-pd}, {Amsf, Afsd, Ansd, A-pd}

Table 1: POS tags used in the mappings for inducing different types of grammatical errors: (1) article misuse, (2) pronoun misuse, (3) incorrect verb suffix and (4) noun-adjective disagreement.

Labelling. The error types in the dataset fall under four types. The process of inducing grammatical errors is more sophisticated than in the case of the spelling error dataset, as the former required understanding of text that goes deeper than syntax.

Errors were induced by identifying a word with a source POS tag and then switching that word for a different inflected form with a target POS tag. The four error types are defined below.

1. **Article Misuse.** If there is a masculine noun with a definite article form, change it to its indefinite form, and vice-versa (e.g. 'синът' → 'сина').
2. **Pronoun Misuse.** If there is a pronoun, change its form:
   - with respect to the object/subject, similar to the use of 'I' and 'me' in English (e.g. 'който' → ' когото').
   - with respect to grammatical gender and/or count (e.g. 'чиито' → 'чийто').
3. **Incorrect Verb Suffix.** If there is a verb in the first person plural form that ends with 'м', append an 'е' (e.g. 'ядем.' → 'ядеме.').
4. **Noun-adjective Disagreement.** If there is a noun-adjective pair, introduce disagreement in terms of count and/or grammatical gender (e.g. 'красива' → 'красив').

The introduction of errors was implemented by defining mappings for source part-of-speech tags to target part-of-speech tags; these tags are provided (organised in one or multiple separate groups for each error) in Table 1. The mappings from source to target tag is generated by computing all possible combinations within each group; for instance, for

[12] https://oscar-project.org

[13] https://bultreebank.org/clark/bulgarian-nlp-pipeline-in-clark-system/

[14] https://www.ims.uni-stuttgart.de/en/research/resources/tools/matetools/

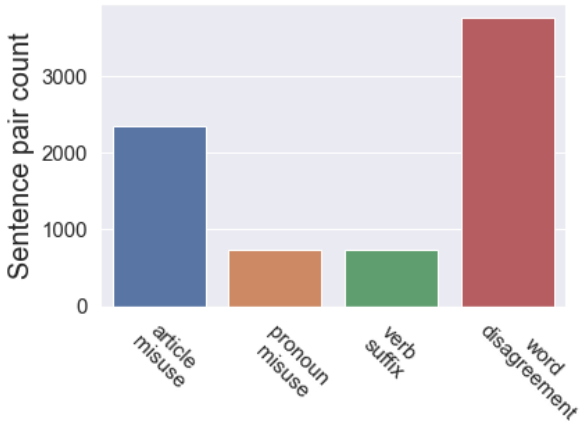

Figure 2: Error type distribution in the grammar error dataset.

article misuse, there is only one group with two POS tags. It follows then that there are only two possible mappings: {Ncmsf → Ncmsh, Ncmsh → Ncmsf}. The second error type contains three groups and overall defines 16 mappings (2 from the first, 2 from the second and 12 from the third). The tags follow the tagging scheme introduced by BulTreeBank (Simov et al., 2004). An exception to the aforementioned process is the third error type which regards incorrect verb suffix misuse. Here, there is only one group with only one relevant POS tag, which is used to identify verbs to append an incorrect suffix to, rather than build mappings from.

The distribution of grammatical error types in our dataset (Figure 2) is influenced by how common the relevant POS tags (corresponding to the error types) are.

4 Fine-tuning mT5 for GEC

4.1 Dataset Selection

For our GEC experiments, we decided to focus on only two of the four error classes we defined, i.e., article and pronoun mistakes, as these are considered to be the most prevalent errors in Bulgarian writing. Filtering the examples based on these error types left us with 3297 pairs. Out of these, we retained only the pairs where neither of the sentences exceeded a 300-character limit, as we consider any sentences longer than that to be anomalous. In the end, 3090 pairs remained. This dataset was utilised for model training and evaluation, whereby subsets with 72%, 18% and 10% of the data were used for training, validation and testing, respectively.

4.2 Model Training

Whereas the original T5 (Raffel et al., 2020) model works exclusively for English, the mT5 model supports multilingual text. Our proposed approach is based on fine-tuning the mT5 model, which has previously demonstrated encouraging performance for GEC in other highly inflected languages (Lytvyn et al., 2023). Specifically, we employed the trained mT5 model available from Huggingface.[15]

During the training stage, an mT5 model takes two sequences, i.e., the source and the target, and learns to transform the first into the second. In our case this would have the source sequence as a sentence with an error and the target sequence would be the same sentence, but corrected. An example is given below.

- Source (erroneous): 'Емануела седна на столът.'
- Target (correct): 'Емануела седна на стола.'

The translation for both is 'Emanuela sat on the chair.' However, in the source sequence, the word 'chair' ('столът') is used in its definite form, instead of indefinite ('стола'). This constitutes a grammatical mistake, as only the subject of the sentence should be used in its definite form.

In order to determine the most optimal values of training hyperpameters, we conducted grid search, whereby the search space was defined based on the hyperparameter values below.

- weight decay rate: {0.1, 0.01, 0.001}
- learning rate: {0.00002, 0.0002, 0.002}
- training batch size: {4, 8}

All 18 hyperparameter combinations were used in fine-tuning the mT5 model for 16 epochs. The process was repeated three times to allow us to take the average over the results, ensuring stable performance.

As can be observed in the visualisation in Figure 3 which presents the validation loss according to the hyperparameter values, a learning rate of 0.002 seems too high, with lower rates yielding better performance. Given this, we performed an experiment to compare the other two learning rates, 0.0002 and 0.00002 (see Figures 6 and 7 in the Appendix). Upon using these two learning rates, it became evident that the former is a better choice. Following this, the final hyperparameter combination that we chose is: learning rate = 0.0002, weight decay = 0.01, batch size = 8. We also decided to fix the

[15] https://huggingface.co/google/mt5-base

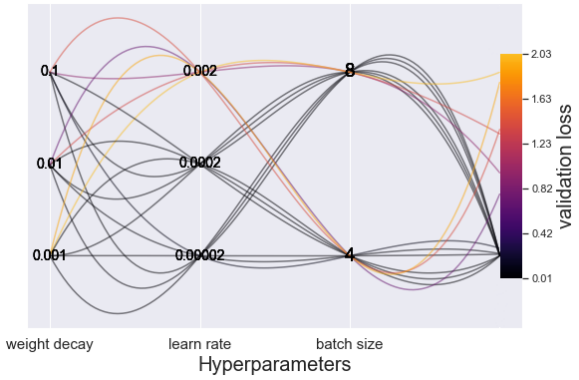

Figure 3: Validation loss for various hyperparameter values: weight decay, learning rate and batch size.

number of epochs to 4, as our experiments showed that this leads to the lowest validation loss.

5 Evaluation

5.1 Dictionary

Our dictionary contains both lemmas and inflected forms, unlike the official one released by The Institute for Bulgarian Language which only contains lemmas. As a means for evaluating its coverage, we randomly selected 20,000 entries from our dictionary. These were then checked against the official dictionary and it was found that only 1292 were present. Within the 93.51% of the missing words, most are inflected versions of base word forms. This only goes to show how existing dictionary resources do not exhibit sufficient coverage for spell-checking tasks.

5.2 GEC Model

Evaluation of the model is performed by comparing its performance on the GEC task with two contemporary large language models (LLMs) that can handle Bulgarian: gpt3.5-turbo[16] and BgGPT.[17] Despite its name which implies that it is based on GPT (Brown et al., 2020), BgGPT is in fact a fine-tuned Mistral model (Jiang et al., 2023). Both of our chosen models for comparison are decoder-only transformers, and rely solely on autoregressive generation. Ideally, evaluation should be performed using other encoder-decoder models; however, no suitable alternatives that can handle Bulgarian were found.

[16] https://platform.openai.com/docs/models/gpt-3-5-turbo

[17] https://huggingface.co/INSAIT-Institute/BgGPT-7B-Instruct-v0.1

Our evaluation involves both quantitative and qualitative comparison, utilising standard metrics and a survey among human participants, respectively.

5.2.1 Quantitative Evaluation

The two LLMs were evaluated based on the commonly used strategy of zero-shot prompting (Rosa et al., 2021), whereby no training examples are provided to the models prior to evaluation. Both models were prompted with each input example from the test set (309 sentence pairs) three times, averaging metric results to account for randomness. Additionally, as prompt engineering has been shown to greatly impact LLM responses (Marvin et al., 2023), two different prompts were utilised.

1. *Correct the mistake: [erroneous sentence]*
2. *Look at the following sentence and rewrite it, fixing any mistakes if there are any: [erroneous sentence]*

The performance of the models, including our fine-tuned mT5 model, is presented in Table 2 in terms of precision, recall and F0.5-score (i.e., F_β score, where $\beta = 0.5$). GEC models are typically evaluated with this F-score setting since the CoNLL-2014 shared task on GEC (Ng et al., 2014), because a lower β score places a higher emphasis on precision, i.e., scoring higher for ensuring predicted tokens are correct, rather than correcting all mistakes.

In this scenario, a true positive (TP) is an erroneous token that has been replaced by its corrected version with respect to the gold reference data. Meanwhile, a false positive (FP) is a correct token being wrongly replaced and a false negative (FN) is an erroneous token that remains unchanged. If a token is erroneous but has been replaced with a token that is not the one specified by the gold standard, it counts both as an FP and an FN.

Our proposed fine-tuned mT5 model outperforms all variants of the contemporary models. Both gpt3.5-turbo and BgGPT scored high on recall, i.e., they corrected a majority of the errors. However, a low precision score implies they tend to over-correct. Their 'corrections' oftentimes do not introduce new errors; they simply reword the source sentence. Nevertheless, they were prompted to only correct errors and rewording runs the risk of changing the sentence semantically.

The proposed fine-tuned mT5 model is not only able to obtain higher recall than both models, but

	Precision	Recall	F0.5-score
gpt3.5-turbo (prompt #1)	37.51 (± 5.40)	60.52 (± 5.45)	39.34 (± 5.45)
BgGPT (prompt #1)	33.07 (± 5.25)	59.87 (± 5.47)	35.03 (± 5.32)
gpt3.5-turbo (prompt #2)	38.62 (± 5.43)	66.02 (± 5.28)	40.74 (± 5.48)
BgGPT (prompt #2)	30.18 (± 5.19)	62.33 (± 5.40)	32.33 (± 5.26)
Fine-tuned mT5 (Ours)	**68.12** (± 5.20)	**68.61** (± 5.17)	**68.18** (± 5.19)

Table 2: Comparison of models for GEC, including 95% confidence intervals.

Figure 4: Proportion of pairwise model preference counts for fine-tuned mT5, gpt3.5-turbo and BgGPT. Rows indicate the winners (preferred model) and columns correspond to their respective opponents.

Figure 5: Bradley-Terry scores from survey rankings for our fine-tuned mT5 model, gpt3.5-turbo and BgGPT with 95% confidence intervals.

also outperforms them across all metrics, especially in terms of F0.5-score. This ensures that the model actively attempts to exclusively correct errors.

5.2.2 Qualitative Evaluation

To assess the correctness, readability and understandability of our proposed solution, we have conducted a survey to compare the performance of the three GEC models from the previous section: BgGPT, gpt3.5-turbo and our fine-tuned mT5 model.

Design. Examples in the test set were used to prompt the proposed solution, as well as BgGPT and gpt3.5-turbo, based on prompt #1, resulting in 309 sentence triplets. The survey included only triplets where all three model predictions are different from one another. In the survey, 13 questions were presented: the first 8 were related to article misuse and the last 5 focussed on pronoun misuse. We refer the reader to Figures 8 and 9 in the Appendix for examples of questions presented to participants as part of our survey.

Results. Overall, 67 Bulgarian native speakers completed the survey. They were recruited by contacting Bulgarian social media groups and AI communities in Bulgaria. Each response served as an indication of pairwise preferences, resulting in 2613 comparisons, provided in Table 3 in the Appendix.

Our fine-tuned mT5 model obtained the highest preference count, with a total of 963. Its preference count proportions, visualised in Figure 4 (57% and 53% vs GPT and BgGPT, respectively) are higher than those of the respective alternative models.

Inspired by the ChatBot Arena[18] (Zheng et al., 2024), we built a Bradley-Terry model to assign a score with confidence intervals to each GEC model based on the responses. As shown in Figure 5, our fine-tuned mT5 model was able to perform better than the contemporary LLMs with a statistically significant difference.

6 Conclusion

This paper presents a state-of-the-art solution for Bulgarian GEC based on the encoder-decoder transformer model mT5, which surpasses LLMs like gpt3.5-turbo and BgGPT. Additionally, we present a contribution in the form of datasets supporting Bulgarian language literacy, including a machine-readable dictionary and two datasets with erroneous-corrected sentence pairs: one for spelling and the other for grammar mistakes.

Future work could investigate additional specialised initiatives regarding the collection of natural learner data from Bulgarian learners. Additionally, language literacy entails punctuation; there is active NLP research in restoring and correcting punctuation in texts (Gravano et al., 2009; Tekir et al., 2023), which can be explored in Bulgarian.

[18] https://chat.lmsys.org

Limitations

Whilst the dictionary collection was evaluated for its coverage, the error correction datasets were generated automatically assuming that the source text is correct.

Our proposed GEC model was trained to identify and correct Bulgarian grammar errors that are based on article and pronoun misuse. Thus, in its current version, it is unlikely to perform well on other types of errors. In addition, the model was not trained to be correct in terms of facts pertaining to people or events, and therefore using the model to generate such content is out-of-scope.

Ethics Statement

The presented datasets and models utilise open-source and publicly available resources (e.g., Wikipedia, OSCAR) that do not contain the names, contact information, addresses, birth dates or other information that can be considered private and/or sensitive.

The survey that we conducted to qualitatively evaluate GEC models did not require users to provide any personal information and no such data was collected for this project.

References

N. Abdul-Rahaman, E. Terentev, and V.E. Arkorful. 2023. The Tertiary Experience: Of Social Integration, Retention and Persistence–A Review. *Public Organization Review, 23(1)*, pages 133–147.

Melania Berbatova and Filip Ivanov. 2023. An Improved Bulgarian Natural Language Processing Pipeline. *Annual of Sofia University St. Kliment Ohridski. Faculty of Mathematics and Informatics*, 110:37–50.

Nadezhda Borisova, Grigor Iliev, and Elena Karashtranova. 2014. On Detecting Noun-Adjective Agreement Errors in Bulgarian Language Using GATE. *Preprint*, arXiv:1411.0588.

Svetla Boytcheva, Boris Velichkov, Gerasim Velchev, and Ivan Koychev. 2020. Automatic Generation of Annotated Corpora of Diagnoses with ICD-10 codes based on Open Data and Linked Open Data. In *2020 15th Conference on Computer Science and Information Systems (FedCSIS)*, pages 163–167.

Chris Brockett, Bill Dolan, and Michael Gamon. 2006. Correcting ESL Errors Using Phrasal SMT Techniques. In *21st International Conference on Computational Linguistics and 44th Annual Meeting of the ACL, Sydney, Australia*, pages 249—-256.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language Models are Few-Shot Learners. *Advances in neural information processing systems*, 33:1877–1901.

Dina C Castro, Mariela M Páez, David K Dickinson, and Ellen Frede. 2011. Promoting Language and Literacy in Young Dual Language Learners: Research, Practice, and Policy. *Child Development*, 5(1):15–21.

Shamil Chollampatt and Hwee Tou Ng. 2018. A Multilayer Convolutional Encoder-Decoder Neural Network for Grammatical Error Correction. In *Proceedings of the AAAI conference on artificial intelligence*, pages 5755–5762.

Denitsa Dimitrova. 2021. Bulgarian Speech Corpora: A Review. In *International CLaDA-BG Conference 2021*, pages 3–58.

Natalia Dolgova and Charles Mueller. 2019. How useful are corpus tools for error correction? insights from learner data. *Journal of English for Academic Purposes*, 39:97–108.

J. Geertzen, T. Alexopoulou, and A. Korhonen. 2013. Automatic linguistic annotation of large scale L2 databases: The EF-Cambridge Open Language Database (EFCamDat). *31st Second Language Research Forum. Somerville, MA: Cascadilla Proceedings Project*, pages 240–254.

Stefan Gerdjikov, Stoyan Mihov, and Vladislav Nenchev. 2013. Extraction of Spelling Variations from Language Structure for Noisy Text Correction. In *2013 12th International Conference on Document Analysis and Recognition*, pages 324–328.

S. Granger, E. Dagneaux, F. Meunier, and M. eds. Paquot. 2009. *International corpus of learner English (Vol. 2)*. Presses Universitaires de Louvain.

Agustin Gravano, Martin Jansche, and Michiel Bacchiani. 2009. Restoring punctuation and capitalization in transcribed speech. In *2009 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 4741–4744.

Roman Grundkiewicz and Marcin Junczys-Dowmunt. 2019. Minimally-Augmented Grammatical Error Correction. In *Proceedings of the 5th Workshop on Noisy User-generated Text (W-NUT 2019)*, pages 357–363.

Roman Grundkiewicz, Marcin Junczys-Dowmunt, and Kenneth Heafield. 2019. Neural Grammatical Error Correction Systems with Unsupervised Pre-training on Synthetic Data. In *Proceedings of the fourteenth workshop on innovative use of NLP for building educational applications*, pages 252–263.

Emi Izumi, Kiyotaka Uchimoto, Toyomi Saiga, Thepchai Supnithi, and Hitoshi Isahara. 2003. Automatic Error Detection in the Japanese Learners' English Spoken Data. In *The Companion Volume to the Proceedings of 41st Annual Meeting of the Association for Computational Linguistics*, pages 145–148.

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

Svetla Koeva, Nikola Obreshkov, and Martin Yalamov. 2020. Natural Language Processing Pipeline to Annotate Bulgarian Legislative Documents. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 6988–6994.

Vladislav Kubon and Martin Plátek. 1994. A Grammar Based Approach to a Grammar Checking of Free Word Order Languages. In *COLING 1994 Volume 2: The 15th International Conference on Computational Linguistics*, pages 906–910.

John Lee and Stephanie Seneff. 2008. Correcting Misuse of Verb Forms. In *46th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, ACL-08: HLT*, pages 174–182.

Jared Lichtarge, Chris Alberti, Shankar Kumar, Noam Shazeer, Niki Parmar, and Simon Tong. 2019. Corpora Generation for Grammatical Error Correction. In *Proceedings of the 2019 Conference of the North*, pages 3291—-3301.

Vasyl Lytvyn, Petro Pukach, Victoria Vysotska, Myroslava Vovk, and Nataliia Kholodna. 2023. Identification and Correction of Grammatical Errors in Ukrainian Texts Based on Machine Learning Technology. *Mathematics*, 11(4):904–923.

Ggaliwango Marvin, Nakayiza Hellen, Daudi Jjingo, and Joyce Nakatumba-Nabende. 2023. Prompt Engineering in Large Language Models. In *International Conference on Data Intelligence and Cognitive Informatics*, pages 387–402.

Jürgen M. Meisel. 2020. Zisa dataset.

National Statistical Institute. 2023. Статистически справочник 'Преброяване 2021' *[Statistical reference book 'Census 2021']*.

Hwee Tou Ng, Siew Mei Wu, Ted Briscoe, Christian Hadiwinoto, Raymond Hendy Susanto, and Christopher Bryant. 2014. The CoNLL-2014 shared task on grammatical error correction. In *Proceedings of the eighteenth conference on computational natural language learning: shared task*, pages 1–14.

OECD. 2023. *PISA 2022 Results (Volume I)*.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. *Journal of machine learning research*, 21(140):1–67.

Gaoqi Rao, Qi Gong, Baolin Zhang, and Endong Xun. 2018. Overview of NLPTEA-2018 share task Chinese grammatical error diagnosis. In *Proceedings of the 5th Workshop on Natural Language Processing Techniques for Educational Applications*, pages 42–51.

Guilherme Moraes Rosa, Ruan Chaves Rodrigues, Roberto de Alencar Lotufo, and Rodrigo Nogueira. 2021. To tune or not to tune? zero-shot models for legal case entailment. In *Proceedings of the Eighteenth International Conference on Artificial Intelligence and Law*, pages 295–300.

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.

K. Simov, P. Osenova, and M. Slavcheva. 2004. BTB-TR03: BulTree-Bank Morphosyntactic Tagset. BTB-TS version 2.0. Technical report, Bulgarian Academy of Sciences.

Kiril Simov, Gergana Popova, and Petya Osenova. 2002. HPSG-based syntactic treebank of Bulgarian (BulTreeBank). *A rainbow of corpora: Corpus linguistics and the languages of the world*, pages 135–142.

Kiril Simov, Alexander Simov, Milen Kouylekov, Krasimira Ivanova, Ilko Grigorov, and Hristo Ganev. 2003. Development of corpora within the CLaRK system: The BulTreeBank project experience. In *Demonstrations*, pages 243–246.

Aiman Solyman, Zhenyu Wang, and Qian Tao. 2019. Proposed Model for Arabic Grammar Error Correction Based on Convolutional Neural Network. In *2019 International Conference on Computer, Control, Electrical, and Electronics Engineering (ICCCEEE)*, pages 1–6.

Aiman Solyman, Marco Zappatore, Wang Zhenyu, Zeinab Mahmoud, Ali Alfatemi, Ashraf Osman Ibrahim, and Lubna Abdelkareim Gabralla. 2023. Optimizing the impact of data augmentation for low-resource grammatical error correction. *Journal of King Saud University-Computer and Information Sciences*, 35(6):101572.

I.M. Suarta, I.K. Suwintana, I.F.P. Sudhana, and N.K.D. Hariyanti. 2017. Employability Skills Required by the 21st Century Workplace: A Literature Review of Labor Market Demand. *International Conference on Technology and Vocational Teachers (ICTVT 2017)*, pages 337–342.

Selma Tekir, Aybüke Güzel, Samet Tenekeci, and Bekir Haman. 2023. Quote Detection: A New Task and

Dataset for NLP. In *Proceedings of the 7th Joint SIGHUM Workshop on Computational Linguistics for Cultural Heritage, Social Sciences, Humanities and Literature*, pages 21–27.

Yu Wang, Yuelin Wang, Kai Dang, Jie Liu, and Zhuo Liu. 2021. A Comprehensive Survey of Grammar Error Correction. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 12(5):1–51.

Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. 2021. mT5: A Massively Multilingual Pre-trained Text-to-Text Transformer. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 483–498.

Zheng Yuan and Ted Briscoe. 2016. Grammatical error correction using neural machine translation. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 380–386.

Wei Zhao, Liang Wang, Kewei Shen, Ruoyu Jia, and Jingming Liu. 2019. Improving Grammatical Error Correction via Pre-Training a Copy-Augmented Architecture with Unlabeled Data. *Preprint*, arXiv:1903.00138.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. 2024. Judging LLM-as-a-Judge with MT-Bench and Chatbot arena. *Advances in Neural Information Processing Systems*, 36:46595–46623.

Appendix

	mT5	GPT	BgGPT
mT5	-	500	463
GPT	371	-	529
BgGPT	408	342	-

Table 3: Pairwise preference counts across the GEC models. Rows indicate the winners (preferred model) and columns correspond to their respective opponents.

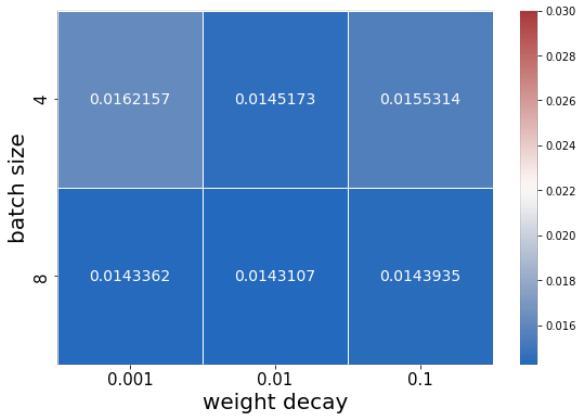

Figure 6: Validation loss obtained by our fine-tuned mT5 model, using a learning rate of 0.0002.

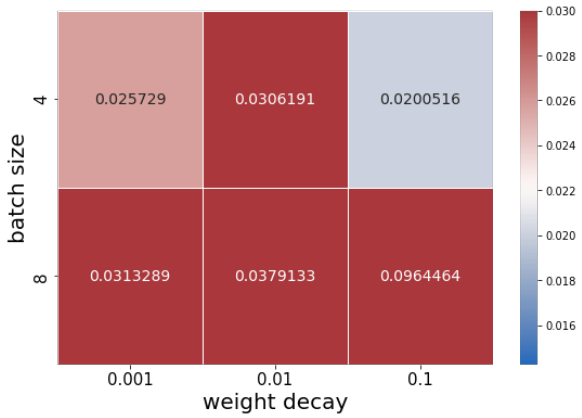

Figure 7: Validation loss obtained by our fine-tuned mT5 model, using a learning rate of 0.00002.

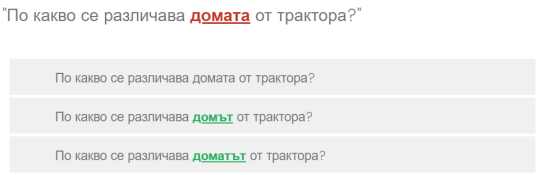

Figure 8: Survey question asking a participant to rank a correction for an article misuse error.

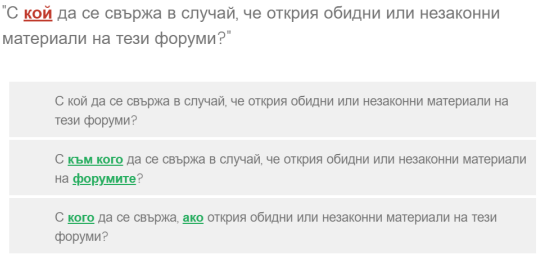

Figure 9: Survey question asking a participant to rank a correction for a pronoun misuse error.