Grammatical Error Correction for Sentence-level Assessment in Language Learning

Anisia Katinskaia[†⋆] and **Roman Yangarber**[†]
University of Helsinki, Finland
[⋆]Department of Computer Science
[†]Department of Digital Humanities
first.last@helsinki.fi

Abstract

The paper presents experiments on using a Grammatical Error Correction (GEC) model to assess the correctness of answers that language learners give to grammar exercises. We empirically check the hypothesis that the GEC model corrects only errors and leaves correct answers unchanged. We perform a test on assessing learner answers in a real but constrained language-learning setup: the learners answer only fill-in-the-blank and multiple-choice exercises. For this purpose, we use ReLCo, a publicly available manually annotated learner dataset in Russian (Katinskaia et al., 2022). In this experiment, we fine-tune a large-scale T5 language model for the GEC task and estimate its performance on the RULEC-GEC dataset (Rozovskaya and Roth, 2019) to compare with top-performing models. We also release an updated version of the RULEC-GEC test set, manually checked by native speakers. Our analysis shows that the GEC model performs reasonably well in detecting erroneous answers to grammar exercises, and potentially can be used in a real learning setting for the best-performing error types. However, it struggles to assess answers which were tagged by human annotators as *alternative-correct* using the aforementioned hypothesis. This is in large part due to a still low recall in correcting errors, and the fact that the GEC model may modify even correct words—it may generate plausible alternatives, which are hard to evaluate against the gold-standard reference.

1 Introduction

Grammatical error correction (GEC) is the task of automatically detecting and correcting grammatical errors in text. Given the recent advancements in Transformer-based GEC models, which have the ability to suggest fluent and grammatically accurate corrections for input sentences, our focus lies in examining their application in language learning settings. One potential application is to check essays written by learners and provide suggestions for corrections—this can be a useful tool for second-language (L2) learners to improve their writing. We are interested in incorporating GEC into an intelligent computer-aided language learning (CALL) system, but in a more constrained scenario: our objective is to evaluate whether a GEC model can be used for automatic assessment of the learner's answers to fill-in-the-blank ("cloze") and multiple-choice (MC) grammar exercises. We assume that this task is comparatively easier than correcting free-text essays, since the number of possible errors in each input sentence is constrained by the number of exercises, these exercises do not change the word order, and our focus is only on grammar. We empirically test the hypothesis: the GEC model can be employed to assess the grammatical correctness of learner answers to grammar exercises, because in an input sentence containing learner answers, the GEC model will fix only tokens with errors—for each erroneous answer it will suggest a correction, and will leave all correct answers unchanged.

In our setting, exercises are generated by Revita, a language learning system, which is used by several hundred L2 learners. These exercises are automatically generated based on a text selected for practice (Katinskaia et al., 2017, 2018). The system has one particular *expected answer* for each exercise—the one found in the original text. When doing an exercise, the learner may insert the expected answer, an error, or an *alternative-correct* answer, which is not expected, but fits the context. The problem can be stated as follows: an unexpected but suitable answer should be recognized as alternative correct, since providing incorrect feedback for valid answers can discourage learners (Katinskaia and Ivanova, 2019; Katinskaia and Yangarber, 2021). For example, in certain sentences, using the present or past tense can be equally acceptable. However, few corpora provide this type of annotation, therefore GEC models are predominantly trained and evaluated using only one reference per instance (Rozovskaya

Proceedings of the 18th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2023), pages 488–502

and Roth, 2021; Bryant et al., 2022).

We use a freely available dataset ReLCo, collected from Revita over several years (Katinskaia et al., 2022). This dataset contains short paragraphs with answers from learners of Russian. The paragraphs include *multiple* answers provided to the same grammar exercises, which were manually checked and tagged as acceptable or erroneous. To the best of our knowledge, this is the only freely available dataset of this type. As a GEC model, we fine-tune a pre-trained monolingual T5 language model (Raffel et al., 2020).

The contributions of this paper are: (1) We show that a GEC model can achieve reasonable performance in assessing erroneous answers for fill-in-the-blank and MC grammar exercises, if we use several top correction hypotheses. We empirically confirm the intuition that a Transformer-based GEC model *cannot* be used for assessing alternative-correct answers since top correction hypotheses can include corrections even for valid words. The lower-ranked hypotheses change the input sentence more freely: include more lexical changes, and more word removals or insertions. (2) We release a new version of the manually corrected RULEC-GEC test set, which, we believe, can improve the evaluation of GEC models in the future. (3) We present the first experiment with ReLCo (Katinskaia et al., 2022), the semi-automatically collected learner data, to train a GEC model. Using ReLCo shows an improvement in GEC performance. (4) We extensively evaluate the performance of our model on the RULEC-GEC (henceforth—RULEC) test set automatically and manually, including an evaluation of several top hypotheses, and show an improvement of $F_{0.5}$ score over the existing state-of-the-art results for Russian. Prior work showed that evaluating GEC output only by automatically comparing it with a single gold-standard reference per sentence results in *under-estimating* the performance (Rozovskaya and Roth, 2021).

The paper is organized as follows. Section 2 covers prior work on the GEC task. Section 3 describes the problem and our approach. Section 4 presents the data for training the GEC model, the training procedure, and the evaluation. Section 5 presents the experiments on assessing learner answers using the trained GEC model. It includes a discussion of results and error analyses. Section 6 presents the conclusions and future work.

2 Related Work

Most current approaches treat GEC as a natural language generation task. It can be formulated as a monolingual translation from incorrect to correct language using various architectures (Yuan and Briscoe, 2016; Junczys-Dowmunt et al., 2018; Chollampatt and Ng, 2018; Yuan et al., 2019; Náplava and Straka, 2019; Grundkiewicz et al., 2019; Zhao et al., 2019; Kaneko et al., 2020). Due to the paucity of annotated training data for GEC, it has become standard practice to generate synthetic data, using various ways of creating erroneous sentences—by back-translation or random token-level transformations (Kiyono et al., 2019), using the history of Wikipedia edits (Lichtarge et al., 2019), confusion sets suggested by spell-checkers (Grundkiewicz et al., 2019; Náplava and Straka, 2019), real error patterns (Choe et al., 2019; Takahashi et al., 2020; Li and He, 2021; Stahlberg and Kumar, 2021), or applying noise to a latent representation of an error-free sentence (Wan et al., 2020). A comparative study of methods of generating synthetic data is presented in (White and Rozovskaya, 2020).

Another approach is text editing—generating a sequence of edits to apply to the incorrect input sentence (Malmi et al., 2019; Stahlberg and Kumar, 2020; Tarnavskyi et al., 2022). In GECTOR (Omelianchuk et al., 2020), the authors develop a set of custom token-level transformations to recover the target text from the source. Editing is faster than generating the whole corrected sentence, but requires constructing many language-specific transformations. More on GEC and existing approaches to the problems and evaluation is reviewed in (Bryant et al., 2022).

A number of papers focus on the actual use of GEC models by language learners. Homma and Komachi (2020) approach the problem of GEC usability as a part of a writing-support system for Japanese, with a focus on inference speed and working with incomplete sentences. Zomer and Frankenberg-Garcia (2021) present a writing-improvement model, which is adapted to the writer's first language (L1). The model's output was evaluated on grammaticality, acceptability, and lexical and syntactic diversity. An Example-Based GEC with a focus on interpretability is introduced in (Kaneko et al., 2022): the model presents to the learners correction results and examples as a base for correction. Takahashi et al. (2022) explore the learners' proficiency-wise evaluation for Quality

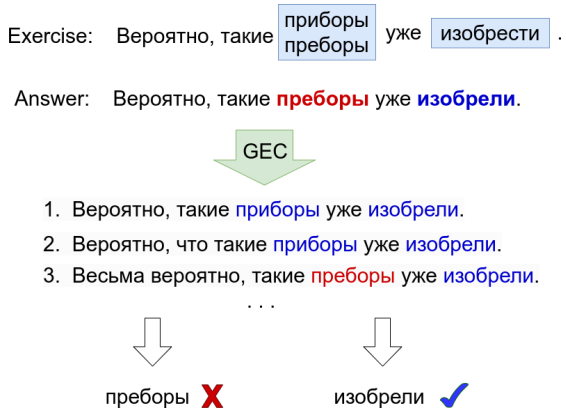

Figure 1: Proposal for how to use a GEC model to check the learner's answers in automatically generated exercises: if an answer was corrected in the *majority* of the top-7 hypotheses, assume it is an error. Otherwise, assume it is correct if it was not altered in the top-3 hypotheses. Red denotes incorrect forms, blue—correct.

Estimation (QE) of GEC.

Several papers on GEC focus on low-resource languages, including Russian (Rozovskaya and Roth, 2019; Katsumata and Komachi, 2020). Náplava and Straka (2019) adapted the approach of Grundkiewicz et al. (2019) for Russian, German, and Czech. Their results on Russian outperformed those of Rozovskaya and Roth (2019) by more than 100% on the $F_{0.5}$ score, but still performed quite poorly compared with other languages. GEC for Russian is shown to be a challenging task, which is explained by the small size of the RULEC corpus. In (Rothe et al., 2021), the biggest multilingual T5 model which was pre-trained on synthetic data and fine-tuned on real data achieved the best performance on Russian among other approaches. Performance was improved by adding to the GEC pipeline a Transformer model for re-ranking the suggested correction edits (Sorokin, 2022).

3 Problem Setup

Our task is to evaluate and provide feedback on the grammatical correctness of answers given by the learner to all grammar exercises (cloze and MC) generated by a CALL system in a sentence. For example, in the sentence in Figure 1, *"Вероятно, такие **приборы** уже **изобретены**."* ("Probably, such **gadgets** have already **been invented**."), the learner receives one MC exercise (*приборы* vs. *преборы*, "gadgets") and a cloze exercise with lemma *"изобрести"* ("to invent"). The MC has only one correct answer, if the exercise is well-designed. In the cloze, the student's answer can be: (1) a definite error; (2) definitely correct, if it matches the expected past passive form *"изобретены"*; or (3) impersonal past tense *"изобрели"*, which is an *acceptable*, slightly different way of saying the same thing (*"Probably someone has already invented such gadgets"*). These alternative corrections can be potentially incorrect in a wider context, but we focus on the context of one sentence to simplify the task.

Dataset	Training	Develop	Test
RULEC	4 980	2 500	5 000
cLang-8 (Ru)	44 830	—	—
ReLCo	8 560	—	7 017

Table 1: Counts of sentence pairs in annotated datasets.

The proposed approach is to use a GEC model whose input is a sentence with all of the learner's answers inserted jointly. This is important, because words chosen by the CALL system for exercises can grammatically depend on each other, and various combinations of answers could be correct, e.g., "gadgets have" vs. "a gadget has." Our conjecture is: if an answer was corrected by a GEC model, it is likely an error; if it was not corrected, it is probably correct. To increase our trust in the model's predictions and address the potential issue of under-corrected errors, we employ a beam search to generate multiple top-ranked hypotheses instead of relying solely on the top-1 correction, see details in Section 5. Previous research by Rozovskaya and Roth (2021) has demonstrated experimentally that lower-ranked hypotheses produced by GEC systems could also be taken into account because they can be qualitatively even better than the top-1 hypotheses, which often suffer from the tendency of GEC systems to under-correct errors due to training with one gold reference per input sentences.

4 GEC Experiments

4.1 Data

To train the GEC model, we use the datasets: RULEC (Rozovskaya and Roth, 2019), Russian cLang-8 (Rothe et al., 2021), and ReLCo (Katinskaia et al., 2022). Dataset statistics are in Table 1. Incorporating the Lang-8 (Tajiri et al., 2012) corpus did not yield a significant improvement, see Table 3. Similar results were shown by Trinh and Rozovskaya (2021), where adding RU-Lang8 to the training data did not improve the results on the

Split	# Errors	# Alternative-Correct
Train	5 642	418
Test	4 316	1 289

Table 2: Number of answers which were manually annotated in ReLCo. Right column: AC learner answers—manually tagged by annotators as "correct", but differ from the expected "reference" answers. Center column: answers which were manually tagged as "errors".

Model	Training Data	$F_{0.5}$
ruT5 large	RULEC	38.10
ruT5 large	RULEC + Lang-8	38.90
ruT5 large	RULEC + cLang-8	39.50
ruT5 large	RULEC + cLang-8 + ReLCo	43.74

Table 3: $F_{0.5}$ scores on the RULEC test data calculated with M^2 scorer. All T5 models reported in this table are not pre-trained on synthetic data.

RULEC test either, although their experiments were conducted using a different model. Therefore, we included the Russian part of the cLang-8 dataset, which is a cleaned version of Lang-8. cLang8 was used only for training. For tuning parameters and analysis of model outputs, we use only the RULEC validation set. The RULEC test set was used for evaluation and comparison with other GEC models.

We split the manually annotated ReLCo into a training and test set. ReLCo consists of short paragraphs, which include learner answers given to grammar exercises. Exercises in the same paragraph can vary depending on the learner's proficiency. The same paragraphs can be practiced by different students or by the same student multiple times, resulting in numerous repeating sentences in the corpus. We ensured that the same sentence never occurs in different data splits. Since we are interested in GEC performance on sentences with multiple acceptable corrections—henceforth, *alternative correct*, or AC—we placed more of such sentences into the test set, see the number of erroneous and AC answers in each data split in Table 2. We also do not want the GEC model to be forced to replace AC answers with expected answers during fine-tuning.

4.2 GEC Model

We use the Text-to-Text Transfer Transformer (T5) model, an encoder-decoder multi-task model that was pre-trained on unsupervised and supervised tasks, with converting each task into a text-to-text format. Rather than the multilingual T5 as in Rothe et al. (2021), we fine-tuned a monolingual Russian T5 model (Raffel et al., 2020).

Rothe et al. (2021) showed that bigger T5 models perform GEC better for all tested languages. We chose a large-size configuration (over 700M parameters), since we cannot run T5 xl or T5 xxl with available resources.

The T5 model is instructed to perform a particular generation task by adding a prefix at the beginning of an input sequence. We conditioned each input sentence by adding the task definition "`improve_grammar`".[1] First, we tried to directly fine-tune the T5 model on the real data, see the results of tuning with several combinations of learner corpora in Table 3. The combination of the RULEC train partition, cLang8, and the ReLCo train partition yields the best F-score and, therefore, it was used in all the following experiments.

Since Rothe et al. (2021) report that the best-performing setup for the T5 model used GEC pre-training on synthetic data, we also (1) pre-train the T5 model on a synthetic dataset until convergence,[2] followed by (2) fine-tuning on the three mentioned datasets.[3] The synthetic data was generated from WMT News Crawl monolingual training data (Bojar et al., 2017) using Aspell confusion sets following (Grundkiewicz et al., 2019). We generated 10M sentences using the same parameters as presented in (Náplava and Straka, 2019). To choose parameters for the fine-tuning on original data, we run hyper-parameter search[4] using Population Based Training (PBT) optimization algorithm (Jaderberg et al., 2017). We set the dropout rate of the T5 model at 0.2, which was found to give the biggest gain in F-score on a validation set. Higher dropout may teach the model to trust the source sentence less and introduce more corrections, as noted previously by Junczys-Dowmunt et al. (2018).

4.3 GEC Evaluation

Given an original sentence with errors (*a source sentence*), the GEC system generates a ranked list of suggested corrections (*hypotheses*). The perfor-

[1] Our implementation is based on Hugging Face.

[2] 3 GPU V100, pre-training for 1.48M steps with batch size = 6, weight decay = 0, learning rate = 5e-5.

[3] The fine-tuned model is available at RuT5_GEC

[4] The best performance was obtained with the following parameters: number of epochs = 2, weight decay = 0.180335, learning rate = 3.83229e-05.

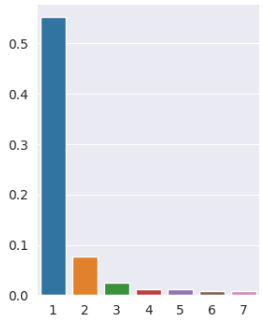

Figure 2: Percentage of hypotheses equal to the gold references in the test set (y-axis) by the rank of the hypotheses (x-label).

System	P	R	$F_{0.5}$
Rozovskaya and Roth (2019)	38.0	7.5	21.0
Trinh and Rozovskaya (2021)	59.1	26.1	47.2
Náplava and Straka (2019)	63.3	27.5	50.2
Rothe, Mallinson, Malmi, Krause, and Severyn (2021) mT5 large	-	-	27.6
Rothe, Mallinson, Malmi, Krause, and Severyn (2021) gT5 xxl	-	-	51.6
Our model	66.6	29.1	52.9

Table 4: Performance of different GEC models for Russian, calculated using M^2 scorer on the RULEC test set.

mance that we discuss next is calculated for the top-1 hypotheses.

Evaluation with M^2 Scorer: Evaluation of all GEC models was done on the RULEC test set using the MaxMatch (M^2) scorer (Dahlmeier and Ng, 2012). It computes GEC performance in terms of phrase-level edits. The results of the evaluation and effect of pre-training and tuning hyper-parameters are shown in Table 5; all reported scores are averaged over 3 runs. A simple pre-processing improvement of the data[5] gives a performance gain, see "data preprocessing" in Table 5. We also have detected some formatting issues and word repetitions in the hypotheses generated by the model for the validation set. Therefore, we run post-processing for the model output, see details in Appendix A. The results of the evaluation after post-processing are in Table 4 and Table 5, and they are on par with the current state of the art.

Evaluation with ERRANT. ERRANT (Bryant et al., 2017; Felice et al., 2016) is a reference-based scorer which measures performance in terms of an edit-based F-score. Unlike the M^2 scorer, it is also able to calculate error type scores at different granularity, e.g., *Replacement* edit or *Replacement:Noun:Case* edit. We use an extension[6] of ERRANT for Russian (Katinskaia et al., 2022). Evaluation of GEC performance using ERRANT was not reported for Russian in the previously published papers. We measured performance with ERRANT on

Model		$F_{0.5}$
ruT5 large	+ RULEC + cLang-8 + ReLCo	43.74
	+ synthetic pre-training	49.62
	+ tuned hyper-parameters	50.83
	+ data preprocessing	51.82
	+ output post-processing	52.94
	+ tested on re-annotated RULEC	55.35
	+ COMET re-ranking	68.19

Table 5: $F_{0.5}$ scores on the RULEC test data calculated using M^2 scorer.

the post-processed output of the best GEC model which we trained, see Table 6. ERRANT's $F_{0.5}$ score for correction is lower (52.1) than $F_{0.5}$ calculated by M^2 scorer (52.9). The same discrepancy between these scores was reported in (Kiyono et al., 2019) for English.

The T5-based GEC model is performing significantly better for replacement errors than for insertion or deletion errors. One possible reason for that can be related to the distribution of error types in the training data: syntactic data was generated mostly by replacing tokens; and in the real learner data, errors were corrected following the principle of minimum correction needed to fix the source sentence, which mostly involves replacing separate words rather than removing or inserting words. ReLCo includes only replacement errors collected from cloze and MC exercises.

Manual Evaluation. Of the total 5 000 top-1 GEC hypotheses generated for the test set, 1 199 were found to be different from both the source sentences and the corresponding gold-standard references. These hypotheses were manually evaluated

[5]Inspection of the GEC model's output on the validation set showed that during inference, the pre-trained and fine-tuned T5 model inserts white spaces into tokens containing the characters [´] ("stress"), or [ë].

Filtering out stress characters and replacing [ë] with [e] is a trivial fix that does not alter the meaning of the text.

[6]RuERRANT

Error type	P	R	$F_{0.5}$	P	R	$F_{0.5}$
	Detection			*Correction*		
Insertion	38.5	8.9	23.2	32.6	7.6	19.6
Replacement	80.9	41.5	67.9	69.6	35.8	58.5
Deletion	36.4	5.3	16.7	24.0	3.2	10.3
Overall	76.0	33.5	60.6	65.3	28.7	52.1

Table 6: Precision, Recall, and F-score measured by ERRANT for span-based error *detection* (left) and span-based error *correction* (right). "Overall" shows performance on all three types of error edits.

by a native-speaking annotator with a degree in teaching Russian and prior annotation experience. The task was to mark whether a sentence is acceptable grammatically. The results showed that 285 of the checked hypotheses can be considered grammatically acceptable. In some cases, the corresponding gold references include typos or uncorrected errors, while in others, GEC hypotheses and the gold reference both present alternative corrections of the source sentence. In addition, 52 hypotheses differ from their gold references only by capitalization, e.g., the first word is not capitalized in a reference, but it is capitalized in the generated hypothesis. The remaining 862 sentences were annotated as indeed ungrammatical. As a final result, the manual evaluation showed that 62.4% of all 5 000 suggested top-1 hypotheses are correct.

Other Hypotheses. We generated 7 hypotheses[7] for each source sentence with beam search decoding. Comparing hypotheses with the references shows that in some cases the GEC model produces a correction which is the same as the reference sentence, but it is not chosen as the top-1 hypothesis. Figure 2 shows the percentage of hypotheses equal to the reference sentences by the rank of the hypotheses. Ranked top 3 include 65.5% of hypotheses equal to the references. More on the manual evaluation of the top-3 hypotheses is in Appendix B.

4.4 RULEC Test Cleaning

Testing various models on the RULEC test set showed that it contains uncorrected errors, ungrammatical corrections, and mistakes in indexing of proposed corrections. Since this impedes assessing the true performance of the models, we undertook a re-annotatation of the data. At this stage, we do not claim that all errors and inconsistencies in the RULEC test set have been fixed.

Annotation was done by three native speakers: two Master's students in linguistics and one expert in teaching Russian. Source sentences were randomly split into two subsets and presented to two annotators, 2.5K sentences each. The annotators could see the original erroneous sentence and its correction (gold reference) proposed in RULEC. The task was to fix the gold reference only if needed, following the minimal-edits principle that results in a grammatically correct reference sentence. The third annotator checked all 5K source sentences and the proposed corrections. Due to limited resources, we could not involve more annotators to correct source sentences without seeing the gold references, or to get more corrections per source sentence. We measured the agreement between the last annotator and the two annotators in the first phase of correction: average agreement is 87%. Most disagreements relate to punctuation, and were resolved by the final annotator.

We calculated our GEC model performance (with output post-processing) on the corrected RULEC test set, see the last row in Table 5. The $F_{0.5}$ score increases to 55.4, which is above the current state-of-the-art results for Russian. A re-annotated test set allowed us to evaluate more realistically the corrections which were attempted by the model, though many errors are still left uncorrected. The updated test set is released in M^2 format.[8]

5 GEC for Evaluating Learner Answers

The following task is to evaluate whether a GEC model can be directly used for assessing learners' answers in a CALL system.

Evaluation. We generated 7 hypotheses for each sentence in the ReLCo test set. The source sentences did not need pre-processing. We applied a post-processing step to filter out 874 hypotheses containing word repetitions, following the method used for the RULEC test set: a filtered-out hypothesis is replaced with its source sentence.

Next, we describe the procedure for checking learner answers based on the suggested corrections. We define the word inserted by the learner as an answer to an exercise as the *target* word. Firstly, we align the suggested GEC hypotheses with the corresponding source sentences. Then, for each

[7] This number of hypotheses is the maximum we can generate with resources available to us.

[8] RULEC-GEC test updated

Answer type	# of answers	P	R	$F_{0.5}$	F_1	Acc.	P	R	$F_{0.5}$	F_1	Acc.	P	R	$F_{0.5}$	F_1	Acc.
		top-3					*all*					*top-3 & re-ranked*				
Gram. error	4 316	89.5	81.9	**87.7**	83.7	-	87.0	87.8	87.1	87.6	-	82.9	90.9	84.4	**88.8**	-
AC	1 289	52.4	67.6	54.8	63.0	-	57.2	55.6	56.9	55.9	-	55.5	72.1	**58.2**	**67.1**	-
Hard AC	206	-	-	-	-	55.3	-	-	-	-	40.8	-	-	-	-	**59.2**

Table 7: Results of estimating the correctness of learners' answers using GEC hypotheses. AC denotes answers which were manually tagged as correct. Hard AC denotes AC answers with the highest disagreement rate among annotators, performance score is accuracy because all instances belong to one class. The best scores are in bold. *top-3*—an answer is considered correct if it is unchanged in all top 3 hypotheses; *all*—an answer is unchanged in all 7 hypotheses; *top-3 & re-ranked*—an answer is unchanged in top-3 hypotheses after re-ranking with COMET score.

target word, we follow the steps:

1. Check whether a target word was corrected by the *majority* of the suggested hypotheses.
2. If corrected by majority, the target word is classified as a grammatical error.
3. Otherwise, check whether the target word was left unchanged in *all* top 3 hypotheses.
4. If not corrected in all top 3 hypotheses, it is potentially an alternative correct answer.
5. Else it is classified as an error.

We chose to evaluate the top-3 hypotheses because previous testing on RULEC showed that they had the highest quality among all generated hypotheses. The results of evaluating grammatical correctness of answers using this algorithm are presented in Table 7, see the third column marked *"top-3"*. Besides $F_{0.5}$, we report the F_1 score: for language learning, it is important not only to provide valid corrections (low false positives) but also not to silently miss errors (low false negatives). Examining multiple hypotheses allows us to improve the precision of detecting AC and the recall of detecting errors. We experimented with modifying steps (3) and (4) by requiring the target word to remain unchanged in all seven suggested hypotheses (see column *"all"* in Table 7). Furthermore, we compared performance on AC answers with the highest disagreement rate among annotators, referred to as "Hard AC" in Table 7. The table presents the performance measured in accuracy, which indicates how many Hard AC answers are recognised as correct.

Re-ranking. One of the problems with using all hypotheses, or only the top-N, is that some of the hypotheses can include more uncorrected errors or can differ significantly from the source sentence lexically and syntactically. For this reason, we experiment with several methods for scoring and re-ranking hypotheses, e.g., using LM scores, the number of errors detected by a GED model, VERNet (Liu et al., 2021), Discriminative re-ranking (Lee et al., 2021), OpenAI's GPT-3.5 model[9] as re-ranker, etc. We test them on RULEC and choose COMET as the best-performing score. Different methods allow increasing precision or recall which, depending on the use case, can be beneficial.

COMET metric[10] for MT evaluation (Stewart et al., 2020; Rei et al., 2020) exploits information from both the source sentence and the reference in order to evaluate the quality of an MT hypothesis. Unlike re-ranking methods which are not using any information about references, COMET allowed to get significant improvement, see performance of a GEC model evaluated after re-ranking with COMET in Table 5. This is the first application of this metric to GEC.

Table 7 (column *"top-3 & re-ranked"*) shows results of assessing learner answers after re-ranking hypotheses according to their COMET score. Using top-3 hypotheses and re-ranking shows the best scores for assessing learners answers overall.

5.1 Error Analysis

We separately analyzed the GEC model's performance in assessing alternative correct answers and erroneous answers.

Alternative Correct (AC). Table 8 shows the accuracy of the GEC model on 14 different types of AC answers in the test data, which were annotated manually. The notation *"Tense: past/present"* means that the expected answer was in the past tense, the learner's answer was in the present, and both forms are acceptable in the context. Performance significantly varies across different types, which should be considered when utilizing GEC

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

[10]The model used is `Unbabel/wmt22-comet-da`

AC category	%	AC category	%
Tense: past/present	85.0	Tense: past/fut.	56.3
Preposition	70.8	Verb: transgr./past	55.6
Number: plur./sing.	68.2	Case: gen./accus.	52.9
Number: sing./plur.	67.2	Adj.: short/full	52.5
Tense: present/past	66.9	Aspect: perf./imperf.	48.7
Tense: fut./past	66.7	Case: instruct./nom.	33.3
Aspect: imperf./perf.	66.4	Case: accus./loc.	31.5

Table 8: Accuracy on estimating AC answers by the GEC model for different categories. Notation "past/pres." means that the learner replaced the past tense form with the present tense; "transgr." denotes transgressive.

for assessing learner answers.

We found that sometimes the GEC model proposes to correct AC answers by words with similar spelling but different meaning that are not relevant in the context, e.g., "бесплотны" (*"ethereal"*) is corrected as "бесплатны" (*"free of charge"*); "от вора" (*"from a thief"*) is corrected as "от ворот" (*"from the gate"*). It especially relates to rare words, e.g., "калорифер" (*"heater"*) replaced with "калории" (*"calories"*). In many cases, the GEC model indeed does not change an AC answer, but it frequently proposes the expected correct answer as a correction, e.g., the top-2 suggestions (Output 1 and 2) in Table 9 include both "смотри" and "посмотри".[11] For more examples, see Appendix D.

Potentially, GEC may be used only for the best-performing types, while for other types, we might need to train separate models. We could provide a learner with 2-3 top corrections suggested by the model and, if it is possible, involve a teacher in a final assessment step.

Errors. One of the detected problems relates to a mismatch between annotation and evaluation conditions: learners' answers in ReLCo were annotated within the context of a paragraph, while we have run GEC evaluation of separate sentences. Therefore, some answers, which are erroneous within a paragraph but not a sentence, were not detected as errors by the model. We run a preliminary evaluation by providing the model with whole paragraphs as input, instead of sentences. Some longer paragraphs have to be pruned to 100 tokens.[12] Performance drops in terms of recall for error detection, though precision increases, especially for a setting

[11] "Look" in imperfect and perfect aspect, respectively.

[12] Due to technical limitations, the input sentence length for beam search cannot exceed 100 tokens.

Source:	на привокзальных площади
Output:	на привокзальных площадях
Expected:	на привокзальной площади
	(*at station square*)
Source:	Он из-за этих документы отвечает.
Output:	Он из-за этих документов отвечает.
Expected:	Он за эти документы отвечает.
	(*He is responsible for these documents.*)
Source:	во перерыве между забегами
Output:	во время перевыва между забегами
Expected:	в перерыве между забегами
	(*during the break between runs*)
Source:	Да ты под переплетом смотри
Output 1:	Да ты под переплетом посмотри
Output 2:	Да ты под переплетом смотри
Expected:	Да ты под переплетом посмотри
	(*Why don't you look under the book cover*)

Table 9: Examples of some source phrases ("Source") with learners' answers (underlined) which were corrected by the model ("Output"). "Expected" shows which answers were expected by Revita CALL system. Red denotes incorrect answers, blue—correct.

with re-ranking. See more details in Appendix C. Paragraph-level assessment needs more investigation in future work.

Another issue is that the GEC model is not informed which word is a target of an exercise. In the first example in Table 9, only the underlined word "привокзальных" (*"near railway station"*) was provided as an answer which is an incorrect plural form in the context. However, the model corrected the noun "площади" (*"square"*) from singular to a plural form. The second example shows issues with reverted word order: the model detects local syntactic relations between the preposition "из-за" (*"because of"*) and the following noun phrase "этих документов" (*"these documents"*), so it puts the noun phrase in genitive case. However, it failed to detect government relations with a head verb "отвечает" (*"is responsible"*), which requires the preposition "за" (*"for"*), not "из-за". The last example shows an issue with checking whether an answer was corrected by the majority of the hypotheses: instead of correcting a preposition "в" ("in"), the model rephrases the whole time expression.

Figure 3 shows the evaluation of detection and correction performance for several error types using ERRANT, on the RULEC and ReLCo test sets. Performance on the two test sets differs drastically on some error types: e.g., spelling, verb aspect, and

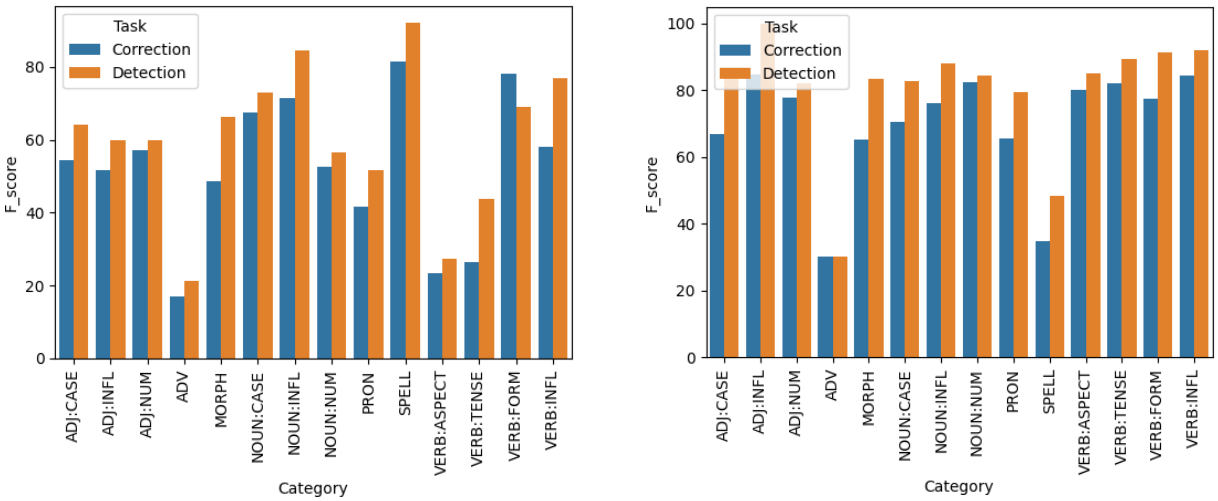

Figure 3: Performance of the GEC model in terms of $F_{0.5}$ score for different error types in the RULEC test set (left) and the ReLCo test set (right).

tense errors. Adverbs and pronouns have low performance in both test sets. All scores for ReLCo are higher, likely because it includes only replacement errors that are better handled by the GEC model. This indicates that the model can potentially be used for detecting errors in cloze exercises with best-performing error types, without providing suggested corrections, since correction performance is lower.

6 Conclusions and Future Work

We present experiments on using Transformer-based GEC models to evaluate the correctness of answers provided by language learners to grammar exercises. To the best of our knowledge, it is the first attempt to directly employ a GEC model for this task. We find that the top-performing GEC model demonstrates the potential to detect and correct errors in user answers provided to fill-in-the-blank and multiple-choice grammar exercises, if we use multiple top hypotheses generated with beam search. However, this approach is less effective for assessing alternative-correct answers. Given the current low recall of the GEC model, there is a high chance of labeling erroneous answers as acceptable. Furthermore, the number of possible alternative corrections proposed by more advanced GEC models can be high, meaning that when the GEC model corrects an answer, it does not necessarily indicate the presence of an error.

The problem of evaluating alternative correct answers is equivalent to the problem of multiple possible corrections for a given error span in GEC. This issue is particularly challenging because GEC models are primarily trained and evaluated using a single reference for each sentence, as discussed in (Rozovskaya and Roth, 2021; Bryant et al., 2022). In our future work, we aim to focus on developing methods for evaluating the suggested corrections by combining reference-based and reference-free scoring approaches.

While GEC is typically approached as a task involving isolated sentences, there have been studies addressing document-level GEC as well (Chollampatt et al., 2019; Yuan and Bryant, 2021). In our experiments, we also focused on assessing grammatical correctness at the sentence level. However, in future work, we plan to investigate the assessment of learner answers within a paragraph. We intend to conduct further research on leveraging large language models to evaluate the acceptability of answers and explore the combination of various re-ranking methods.

Acknowledgements

This work was supported in part by the Academy of Finland, Helsinki Institute for Information Technology (HIIT), BusinessFinland (Grant "Revita", 42560/31/2020), and Tulevaisuusrahasto, the Future Development Fund, Faculty of Arts, University of Helsinki.

7 Ethical Considerations

We use only publicly available resources for all conducted experiments. All annotators were volunteer students who performed the tasks as a part of their studies and received credits for it.

8 Limitations

The current work has a number of limitations to consider.

(A) The paper's experimental design was limited to a single language because we are not aware of any other learner corpora with multiple answers provided to the same exercises.

(B) The described approach to assessing the correctness of learner answers is limited by its design. First, the number of GEC hypotheses to check depends on the GEC model's performance and, potentially, on the language. Second, if a word was not corrected, it can be a false negative error instead of a correct answer. Third, the GEC model can suggest corrections (valid and not valid) even to a correct answer depending on the data it was trained on.

(C) Our approach focuses only on grammatical errors and it does not take into account semantic or pragmatic errors.

(D) Due to limited resources, we were unable to involve more people with prior annotation experience in the re-annotation of the RULEC test set, as well as in the manual verification of hypotheses generated by the GEC model. We acknowledge that the annotation performed by our annotators may not be entirely error-free: the annotators were free to work at their own pace and therefore could potentially rush and make errors themselves. Hence, we do not claim that the re-annotated RULEC test set does not include any inconsistency anymore. We believe that the existing datasets should be thoroughly checked, given the small amount of learner data available for languages other than English, before utilizing them to train and evaluate new models.

(E) Considering the practical use of our GEC model as a component of a CALL system, we find that it can potentially be used in a limited context, i.e., for checking answers provided to cloze and multiple-choice exercises, only for best-performing error types. As for alternative correct answers, even for best-performing categories of answers, a human teacher should verify proposed corrections. We have to underline that learner errors in RULEC-GEC (and especially in any synthetic dataset) can significantly differ from errors made by learners with various backgrounds, native languages, and proficiency levels. We also find that low recall of state-of-the-art GEC models impedes their usage in language learning settings. At the moment, learner answers should be verified by a human teacher.

References

Ondřej Bojar, Rajen Chatterjee, Christian Federmann, Yvette Graham, Barry Haddow, Shujian Huang, Matthias Huck, Philipp Koehn, Qun Liu, Varvara Logacheva, Christof Monz, Matteo Negri, Matt Post, Raphael Rubino, Lucia Specia, and Marco Turchi. 2017. Findings of the 2017 conference on machine translation (WMT17). In *Proceedings of the Second Conference on Machine Translation*, pages 169–214, Copenhagen, Denmark. Association for Computational Linguistics.

Christopher Bryant, Mariano Felice, and Ted Briscoe. 2017. Automatic annotation and evaluation of error types for grammatical error correction. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 793–805, Vancouver, Canada. Association for Computational Linguistics.

Christopher Bryant, Zheng Yuan, Muhammad Reza Qorib, Hannan Cao, Hwee Tou Ng, and Ted Briscoe. 2022. Grammatical error correction: a survey of the state of the art. *arXiv preprint arXiv:2211.05166*.

Yo Joong Choe, Jiyeon Ham, Kyubyong Park, and Yeoil Yoon. 2019. A neural grammatical error correction system built on better pre-training and sequential transfer learning. In *Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 213–227, Florence, Italy. Association for Computational Linguistics.

Shamil Chollampatt and Hwee Tou Ng. 2018. A multilayer convolutional encoder-decoder neural network for grammatical error correction. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence*.

Shamil Chollampatt, Weiqi Wang, and Hwee Tou Ng. 2019. Cross-sentence grammatical error correction. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 435–445, Florence, Italy. Association for Computational Linguistics.

Daniel Dahlmeier and Hwee Tou Ng. 2012. Better evaluation for grammatical error correction. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 568–572, Montréal, Canada. Association for Computational Linguistics.

Mariano Felice, Christopher Bryant, and Ted Briscoe. 2016. Automatic extraction of learner errors in ESL sentences using linguistically enhanced alignments. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 825–835, Osaka, Japan. The COLING 2016 Organizing Committee.

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, Florence, Italy. Association for Computational Linguistics.

Hiroki Homma and Mamoru Komachi. 2020. Non-autoregressive grammatical error correction toward a writing support system. In *Proceedings of the 6th Workshop on Natural Language Processing Techniques for Educational Applications*, pages 1–10, Suzhou, China. Association for Computational Linguistics.

Max Jaderberg, Valentin Dalibard, Simon Osindero, Wojciech M Czarnecki, Jeff Donahue, Ali Razavi, Oriol Vinyals, Tim Green, Iain Dunning, Karen Simonyan, et al. 2017. Population based training of neural networks. *arXiv preprint arXiv:1711.09846*.

Marcin Junczys-Dowmunt, Roman Grundkiewicz, Shubha Guha, and Kenneth Heafield. 2018. Approaching neural grammatical error correction as a low-resource machine translation task. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 595–606, New Orleans, Louisiana. Association for Computational Linguistics.

Masahiro Kaneko, Masato Mita, Shun Kiyono, Jun Suzuki, and Kentaro Inui. 2020. Encoder-decoder models can benefit from pre-trained masked language models in grammatical error correction. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4248–4254, Online. Association for Computational Linguistics.

Masahiro Kaneko, Sho Takase, Ayana Niwa, and Naoaki Okazaki. 2022. Interpretability for language learners using example-based grammatical error correction. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7176–7187, Dublin, Ireland. Association for Computational Linguistics.

Anisia Katinskaia and Sardana Ivanova. 2019. Multiple admissibility: Judging grammaticality using unlabeled data in language learning. In *Proceedings of the 7th Workshop on Balto-Slavic Natural Language Processing*, pages 12–22, Florence, Italy. Association for Computational Linguistics.

Anisia Katinskaia, Maria Lebedeva, Jue Hou, and Roman Yangarber. 2022. Semi-automatically annotated learner corpus for Russian. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 832–839, Marseille, France. European Language Resources Association.

Anisia Katinskaia, Javad Nouri, and Roman Yangarber. 2017. Revita: a system for language learning and supporting endangered languages. In *Proceedings of the joint workshop on NLP for Computer Assisted Language Learning and NLP for Language Acquisition*, pages 27–35, Gothenburg, Sweden. LiU Electronic Press.

Anisia Katinskaia, Javad Nouri, and Roman Yangarber. 2018. Revita: a language-learning platform at the intersection of ITS and CALL. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Anisia Katinskaia and Roman Yangarber. 2021. Assessing grammatical correctness in language learning. In *Proceedings of the 16th Workshop on Innovative Use of NLP for Building Educational Applications*, pages 135–146.

Satoru Katsumata and Mamoru Komachi. 2020. Stronger baselines for grammatical error correction using a pretrained encoder-decoder model. In *Proceedings of the 1st Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing*, pages 827–832, Suzhou, China. Association for Computational Linguistics.

Shun Kiyono, Jun Suzuki, Masato Mita, Tomoya Mizumoto, and Kentaro Inui. 2019. An empirical study of incorporating pseudo data into grammatical error correction. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1236–1242, Hong Kong, China. Association for Computational Linguistics.

Ann Lee, Michael Auli, and Marc'Aurelio Ranzato. 2021. Discriminative reranking for neural machine translation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 7250–7264, Online. Association for Computational Linguistics.

Xia Li and Junyi He. 2021. Data augmentation of incorporating real error patterns and linguistic knowledge for grammatical error correction. In *Proceedings of the 25th Conference on Computational Natural Language Learning*, pages 223–233, Online. Association for Computational Linguistics.

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 American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 3291–3301, Minneapolis, Minnesota. Association for Computational Linguistics.

Zhenghao Liu, Xiaoyuan Yi, Maosong Sun, Liner Yang, and Tat-Seng Chua. 2021. Neural quality estimation with multiple hypotheses for grammatical error correction. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for*

Computational Linguistics: Human Language Technologies, pages 5441–5452, Online. Association for Computational Linguistics.

Eric Malmi, Sebastian Krause, Sascha Rothe, Daniil Mirylenka, and Aliaksei Severyn. 2019. Encode, tag, realize: High-precision text editing. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5054–5065, Hong Kong, China. Association for Computational Linguistics.

Jakub Náplava and Milan Straka. 2019. Grammatical error correction in low-resource scenarios. In *Proceedings of the 5th Workshop on Noisy User-generated Text (W-NUT 2019)*, pages 346–356, Hong Kong, China. Association for Computational Linguistics.

Kostiantyn Omelianchuk, Vitaliy Atrasevych, Artem Chernodub, and Oleksandr Skurzhanskyi. 2020. GECToR – grammatical error correction: Tag, not rewrite. In *Proceedings of the Fifteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 163–170, Seattle, WA, USA → Online. Association for Computational Linguistics.

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.

Ricardo Rei, Craig Stewart, Ana C Farinha, and Alon Lavie. 2020. COMET: A neural framework for MT evaluation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2685–2702, Online. Association for Computational Linguistics.

Sascha Rothe, Jonathan Mallinson, Eric Malmi, Sebastian Krause, and Aliaksei Severyn. 2021. A simple recipe for multilingual grammatical error correction. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 702–707, Online. Association for Computational Linguistics.

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.

Alla Rozovskaya and Dan Roth. 2021. How good (really) are grammatical error correction systems? In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 2686–2698, Online. Association for Computational Linguistics.

Alexey Sorokin. 2022. Improved grammatical error correction by ranking elementary edits. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 11416–11429, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Felix Stahlberg and Shankar Kumar. 2020. Seq2Edits: Sequence transduction using span-level edit operations. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5147–5159, Online. Association for Computational Linguistics.

Felix Stahlberg and Shankar Kumar. 2021. Synthetic data generation for grammatical error correction with tagged corruption models. In *Proceedings of the 16th Workshop on Innovative Use of NLP for Building Educational Applications*, pages 37–47, Online. Association for Computational Linguistics.

Craig Stewart, Ricardo Rei, Catarina Farinha, and Alon Lavie. 2020. COMET - deploying a new state-of-the-art MT evaluation metric in production. In *Proceedings of the 14th Conference of the Association for Machine Translation in the Americas (Volume 2: User Track)*, pages 78–109, Virtual. Association for Machine Translation in the Americas.

Toshikazu Tajiri, Mamoru Komachi, and Yuji Matsumoto. 2012. Tense and aspect error correction for ESL learners using global context. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 198–202, Jeju Island, Korea. Association for Computational Linguistics.

Yujin Takahashi, Masahiro Kaneko, Masato Mita, and Mamoru Komachi. 2022. ProQE: Proficiency-wise quality estimation dataset for grammatical error correction. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 5994–6000, Marseille, France. European Language Resources Association.

Yujin Takahashi, Satoru Katsumata, and Mamoru Komachi. 2020. Grammatical error correction using pseudo learner corpus considering learner's error tendency. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop*, pages 27–32, Online. Association for Computational Linguistics.

Maksym Tarnavskyi, Artem Chernodub, and Kostiantyn Omelianchuk. 2022. Ensembling and knowledge distilling of large sequence taggers for grammatical error correction. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3842–3852, Dublin, Ireland. Association for Computational Linguistics.

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

Zhaohong Wan, Xiaojun Wan, and Wenguang Wang. 2020. Improving grammatical error correction with data augmentation by editing latent representation. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 2202–2212, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Max White and Alla Rozovskaya. 2020. A comparative study of synthetic data generation methods for grammatical error correction. In *Proceedings of the Fifteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 198–208, Seattle, WA, USA → Online. Association for Computational Linguistics.

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, San Diego, California. Association for Computational Linguistics.

Zheng Yuan and Christopher Bryant. 2021. Document-level grammatical error correction. In *Proceedings of the 16th Workshop on Innovative Use of NLP for Building Educational Applications*, pages 75–84, Online. Association for Computational Linguistics.

Zheng Yuan, Felix Stahlberg, Marek Rei, Bill Byrne, and Helen Yannakoudakis. 2019. Neural and FST-based approaches to grammatical error correction. In *Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 228–239, Florence, Italy. Association for Computational Linguistics.

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. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 156–165, Minneapolis, Minnesota. Association for Computational Linguistics.

Gustavo Zomer and Ana Frankenberg-Garcia. 2021. Beyond grammatical error correction: Improving L1-influenced research writing in English using pre-trained encoder-decoder models. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 2534–2540, Punta Cana, Dominican Republic. Association for Computational Linguistics.

A Cleaning Model's Output

We have discovered two issues in the hypotheses generated by the GEC model for the validation set. One is extra white spaces in front of hyphenated suffixes added to numbers, e.g., "25 -го апреля" ("*on the 25th of April*") instead of "25-го апреля". These extra spaces were removed. Another issue relates to corrections of some short sentences (1-5 words): the generated hypotheses have repeating tokens. It is especially relevant to incomplete sentences in the test set which end with a semicolon, e.g., "Рай :" ("*Heaven :*"). The model is either trying to continue these sentences or just repeating the same word. We have detected all hypotheses for which source sentences were shorter than 6 words (without punctuation) and which include repetitions and replaced them with the source sentences as if they were not corrected by the model at all, in total 44 sentences.

B Manual Evaluation of Top Hypotheses

We have picked top-3 hypotheses for 100 randomly sampled source sentences from the RULEC test set. These hypotheses were manually evaluated by a native speaker on the following aspect: whether the second-ranked and the third-ranked hypotheses improve the corrections suggested in the top-1 hypothesis or whether the quality of corrections degrades. Manual evaluation has shown that for 58% of checked sentences, the quality only improves with more hypotheses.

C Paragraph Correction

Due to technical limitations, the GEC model input length cannot exceed 100 tokens. Therefore, to run a preliminary evaluation with whole paragraphs as input, instead of sentences, we had to prune the longest paragraphs to 100 tokens. This leads to losing 107 learner answers. Regarding assessing and detecting grammatical errors, recall drops and precision increases, especially for a setting with re-ranking, see Table **??**. As a result, this leads to lower precision for AC answers, since more errors are not corrected. We find several reasons for this decrease in error recall. First of all, the GEC model was pre-trained and fine-tuned on sentences. For example, it corrects erroneous "уже из конца недели" to "уже в конце недели" ("*already at the end of the week*") only if this error is in a separate sentence. However, the error is not corrected if the model gets as input a three-sentence paragraph, with this error in the second sentence. Another issue relates to pruning paragraphs which leads to incomplete sentences and broken syntactic relations between words. Paragraph-level assessment requires more research and training GEC models on a wider context, though there are few available datasets for this task.

D Examples

Table 10 presents more examples where the GEC model generates multiple valid corrections in the same context.

Source:	сел рядом на скамеечку
Output 1:	сел рядом на скамеечку
Output 2:	сел рядом на скамеечке
Output 3:	сел рядом на скамейке
Expected:	сел рядом на скамеечке
	(*sat on a bench nearby*)
Source:	автор работы обнаружил...
Output 1:	автор работы обнаружил...
Output 2:	авторы работы обнаружили...
Expected:	авторы работы обнаружили...
	(*the authors found...*)
Source:	Коврин был уже мертвым, когда...
Output 1:	Коврин был уже мертв, когда...
Output 2:	Коврин был уже мертвым, когда...
Expected:	Коврин был уже мертв, когда...
	(*Kovrin was already dead, when...*)
Source:	...хохотал он
Output 1:	...хохотал он
Output 2:	...расхохотался он
Expected:	...хохочет он
	(*...he laughed*)
Source:	Большинство заданий выполняется быстро
Output 1:	Большинство заданий выполняются быстро
Output 2:	Большинство заданий выполняется быстро
Expected:	Большинство заданий выполняются быстро
	(*Most tasks are done fast*)
Source:	просто удивляюсь и не верю: ты ли это?
Output 1:	просто удивляюсь и не верю: ты ли это?
Output 2:	просто удивляешься и не веришь: ты ли это?
Expected:	просто удивляешься и не веришь: ты ли это?
	(*just surprised and can't believe, is it you?*)
Source:	сторожей, подобных этому, я не увидел
Output 1:	сторожей, подобных этому, я не увидел
Output 2:	сторожей, подобных этому, я не видел
Expected:	сторожей, подобных этому, я не увидал
	(*I have not seen watchmen like this*)
Source:	проявления мстительности и вредительство
Output 1:	проявления мстительности и вредительство
Output 2:	проявления мстительности и вредительства
Expected:	проявления мстительности и вредительства
	(*manifestations of revenge and wrecking*)
Source:	ребенок трогательно погладил моих собак
Output 1:	ребенок трогательно погладил моих собак
Output 2:	ребенок трогательно поглаживал моих собак
Expected:	ребенок трогательно гладил моих собак
	(*the child touchingly stroked my dogs*)
Source:	как сформировался этот регион
Output 1:	как сформировался этот регион
Output 2:	как сформирован этот регион
Expected:	как формировался этот регион
	(*how this region was formed*)

Table 10: Examples of some source phrases ("Source") with learners' AC answers (blue underlined) which were corrected by the model. "Output 1" and "Output 2" denote the top-2 model's corrections. "Expected" shows which answers were expected by Revita CALL system.