Low-Resource Grammatical Error Correction: Selective Data Augmentation with Round-Trip Machine Translation

Frank Palma Gomez
Boston University
frankpalma12@gmail.com

Alla Rozovskaya
City University of New York
arozovskaya@qc.cuny.edu

Abstract

Supervised state-of-the-art methods for grammatical error correction require large amounts of parallel data for training. Due to lack of gold-labeled data, techniques that create synthetic training data have become popular. We show that models trained on synthetic data tend to correct a limited range of grammar and spelling mistakes that involve character-level changes, but perform poorly on (more complex) phenomena that require word-level changes. We propose to address the performance gap on such errors by generating synthetic data through selective data augmentation via round-trip machine translation. We show that the proposed technique, SeLex-RT, is capable of generating mistakes that are similar to those observed with language learners. Using the approach with two types of state-of-the-art learning frameworks and two low-resource languages (Russian and Ukrainian), we achieve substantial improvements, compared to training on synthetic data produced with standard techniques. Analysis of the output reveals that models trained on data noisified with the SeLex-RT approach are capable of making word-level changes and correct lexical errors common with language learners.[1]

1 Introduction

Grammatical Error Correction (GEC) is the task of detecting and correcting mistakes in text. Supervised state-of-the-art approaches to GEC require large amounts of training data in the form of sentence examples with errors and their corrected counterparts. Because hand-labeled data is expensive to obtain, it is common practice to use monolingual data with *synthetic noise* for pre-training. The models are further finetuned on gold learner data. The use of synthetic data is essential to obtaining good performance, especially when the language is low-resource and has a limited amount of gold-labeled training data (Flachs et al., 2021). However, existing methods do not generate sufficiently diverse synthetic errors (Stahlberg and Kumar, 2024).

We show that low-resource state-of-the-art strategies for generating synthetic data mainly produce errors in spelling and inflectional morphology, and models trained on such synthetic noise fail to address errors related to overall fluency that are known to pose challenges even to high-proficiency learners (Yasunaga et al., 2021; Choshen and Abend, 2018b). *Fluency* is used to refer to edits that make the original text native-sounding; these edits typically go beyond grammaticality and may include a change for a more preferred word order or a better lexical choice (Napoles et al., 2017).

In this work, we focus on correcting *lexical errors*. Many of these errors are known to arise from first-language interference (Leacock et al., 2010; Dahlmeier and Ng, 2011; Rozovskaya et al., 2017). Moreover, studies in second language acquisition reveal that learners may generate a sentence in the second language by *translating* it from their native language (Derakhshan and Karimi, 2015), thereby incorrectly transferring structures and expressions into the second language. A lexical error may occur when a learner picks an incorrect but *semantically-related* translation for an ambiguous word with several related meanings in their native language.

Using this observation, we hypothesize that lexical challenges, such as choosing an incorrect word in the second language, will also manifest themselves in the machine translation output, as back-translated words that are semantically close to the target. We propose to address such errors using select substitutions obtained from round-trip machine translation. Indeed, we show that the approach with select substitutions is capable of generating a wide range of mistakes that non-native speakers make, and is particularly good at producing confusions similar to lexical errors common among non-native speakers. We refer to this approach as *SeLex-RT*.

We start with a monolingual corpus of language l for which we wish to build a GEC model. The sen-

[1] The data is available at https://github.com/arozovskaya/Low-Resource-GEC-SeLex-RT

Findings of the Association for Computational Linguistics: ACL 2025, pages 25749–25770
July 27 - August 1, 2025 ©2025 Association for Computational Linguistics

tences are translated into another language (pivot), and then back into *l*. The original sentence is token-aligned with its round-trip translation, and the alignments are used to create confusions resulting from imperfect translations. In contrast to other approaches that employ round-trip translation (Lichtarge et al., 2019; Kementchedjhieva and Søgaard., 2023), we (1) do not make use of the entire resulting back-translated sentences, but only generate *targeted confusion sets* of relevant errors that are used to corrupt the data; and (2) generate multiple translation hypotheses in each direction. We demonstrate that both of these innovations are crucial for obtaining a diverse set of high-quality synthetic errors (see Section 6.1).

We present experiments on two low-resource languages for GEC, Russian and Ukrainian, with additional analyses on Russian. We show that only 5%-12% of the lexical errors observed in gold learner data are represented in the data produced with the standard synthetic methods. In contrast, SeLex-RT is capable of generating synthetic data that represents over 50% of learner lexical errors. Experimental results demonstrate substantial performance gains on correcting lexical errors, compared to training with standard synthetic data.

The paper makes the following contributions: (1) Using two Russian benchmarks, we show that the majority of mistakes identified by state-of-the-art supervised GEC models are in spelling and inflectional morphology; (2) We propose *SeLex-RT*, a novel method that uses select substitutions from round-trip translations and generate synthetic errors that involve replacements of unrelated but semantically similar words and complements existing approaches; (3) We use *SeLex-RT* to generate synthetic data for two low-resource languages, and substantially improve over standard methods; additional analysis on Russian reveals that *SeLex-RT* is particularly beneficial for correcting lexical errors.

2 Background

Supervised approaches to GEC can be broken down into sequence-to-sequence (seq2seq) (Chollampatt and Ng, 2018; Yuan and Briscoe, 2016; Grundkiewicz et al., 2019; Grundkiewicz and Junczys-Dowmunt, 2019; Kiyono et al., 2019; Zhao et al., 2019; Ji et al., 2017; Katsumata and Komachi., 2019; Xie et al., 2018; Qorib et al., 2024), and sequence-to-editing (Omelianchuk et al., 2020; Awasthi et al., 2019; Tarnavskyi et al., 2022). Both achieve state-of-the-art performance on English GEC. Recent works also achieve strong results with edit ranking (Sorokin, 2022) or ensembling (Omelianchuk et al., 2024; Qorib and Ng, 2023). Other works have made additional advancements on English GEC, but they rely on large amounts of hand-labeled data that is not available for other languages (Sun and Wang, 2022; Lai et al., 2022; Bout et al., 2023).

The edit-based framework (Stahlberg and Kumar, 2020; Omelianchuk et al., 2020) was shown to be competitive on English, however, for other languages proved to be less successful (Syvokon and Romanyshyn, 2023), due to the fact that the approach requires language-specific knowledge to develop rules (Bryant et al., 2023). (It is important to note that almost all non-English GEC can be viewed as low-resource, due to scarcity of hand-labeled training data). The seq2seq framework, on the other hand, has shown state-of-the-art performance in non-English GEC (Rothe et al., 2021), and we thus adopt this approach in our work. In the seq2seq approach, GEC is cast as a machine translation task with the erroneous sentences treated as the source and corrected sentences treated as the target. Sentences with real learner errors and their manually-corrected counterparts or sentences from a monolingual corpus with added synthetic noise can be used for training GEC models. Pre-trained language models (PLMs) can be used as a starting point (Kaneko et al., 2020; Malmi et al., 2019; Omelianchuk et al., 2020; Katsumata and Komachi., 2019). We follow Rothe et al. (2021) that successfully applied the approach in multilingual settings, making use of mT5.

Low-resource GEC Compared to English, Russian and Ukrainian are low-resource for GEC, due to the limited amounts of labeled training data (see Table 1). In terms of monolingual data, the mC4 corpus used to pre-train mT5, can be considered as high resource for Russian (3.6 TB of data) but low-resource for Ukrainian (196 GB of data).

3 Learner Datasets

Russian and Ukrainian gold data We use two datasets of Russian learner data manually corrected for errors[2] RULEC-GEC (Alsufieva et al., 2012; Rozovskaya and Roth, 2019) (henceforth RULEC) and RU-Lang8 (Trinh and Rozovskaya, 2021). Both datasets were originally annotated with a single gold reference each (a corrected version produced by an expert), however, the annotations have been en-

[2] Manually corrected learner data is referred to as *gold data*.

Dataset	Partition (sents.)		
	Train	Dev.	Test
RULEC	4,980	2,500	5,000
RU-Lang8	-	1,968	2,444
UA-GEC	32,734	1,506	2,644

Table 1: Russian and Ukrainian gold datasets.

Error group	Percentage (%)	
	RULEC	RU-Lang8
Grammar	39.3	40.0
Orth.	32.7	33.0
Lex./morph.	14.0	13.5
Other	14.0	13.5
Total	5283	3382

Table 2: Learner error distributions in Russian by coarse category. *Other* includes word deletion/insertion, word order, phrase replacement.

riched with two additional references, for a total of three references per sentence (Palma Gomez and Rozovskaya, 2024). We use the enriched benchmarks. For Ukrainian, we use the UA-GEC dataset (Syvokon and Romanyshyn, 2023). We report results on the *Fluency* track (that includes both grammar and fluency edits). Dataset sizes are in Table 1.

Error distributions in learner Russian To understand the distribution of errors in learner data and to perform evaluations by error type (Section 5), we classify gold errors in the Russian data using a tool for Russian (Rozovskaya, 2022).[3] The tool is similar in spirit to ERRANT, developed for English (Bryant et al., 2017), and classifies edits into 24 categories in spelling, punctuation, morphology, and lexical errors. We further group all errors into four broad categories: *grammar*, *orthography*, *lexico-morphology*, and *other* (Table 2). Mistakes in the *grammar* category involve 1-2 character modifications in inflections or a small confusion set (errors on closed-class words). The *orthography* category includes spelling, punctuation, and capitalization mistakes. The *lexico-morphology* category includes lexical mistakes and errors in derivational morphology, i.e. those that involve word formation and go beyond single character misuse. This category is the main focus of our work. Sample learner errors are illustrated in Appendix Table B3.

4 Synthetic Data Methods

Ye et al. (2023) states the following: "the lack of high-quality publicly available data remains a challenge in low resource settings." In this section, we first review existing approaches to generating synthetic data, including the baseline methods used in this work. Next, we introduce the *SeLex-RT* approach, describe how the errors are generated with each method. We then compare the synthetic errors produced with each method with errors found in the learner texts.

4.1 Existing Synthetic Data Methods

Overview of existing methods used as baselines Synthetic data generation methods lie along a continuum from knowledge-lean to knowledge-intensive, depending on the language-specific knowledge and resources required. As baselines, we implement data generation techniques that are general enough to be used across a variety of languages and do not require extensive language-specific knowledge.

Random character-level perturbations (Char) This approach creates synthetic errors in monolingual data, by probabilistically inserting, deleting, or perturbing characters (Kiyono et al., 2019; Grundkiewicz et al., 2019; Flachs et al., 2021; Stahlberg et al., 2019). We refer to it as *Char*.

Spell-based transformations (Spell) It combines character-level perturbations of the *Char* method and token-level perturbations generated from confusions from an open-source spellchecker such as Aspell (Grundkiewicz and Junczys-Dowmunt, 2019). This approach showed state-of-the-art performance in English (Bryant et al., 2019; Grundkiewicz and Junczys-Dowmunt, 2019; Grundkiewicz et al., 2019) and in other languages (Náplava and Straka, 2019).

Morphology-based transformations (Morph) The third baseline method is based on the approach in Choe et al. (2019), and derives token-level confusions from morphological transformations. Originally applied in English, the approach also showed good results on other languages with rich morphology (Flachs et al., 2021). The original method of Choe et al. (2019) combined morphological transformations with patterns mined from the development data for English. Flachs et al. (2021) implement a similar approach, but only use morphological transformations, without the patterns, in four languages, including Russian. We follow Flachs et al. (2021) and only use morphological transformations.

We use the three synthetic methods above – *Char*, *Spell*, and *Morph* – as the baselines in our experi-

[3] We are not aware of a similar tool for Ukrainian.

ments (see Section 5).

Other synthetic methods not used in this work Several other methods of generating synthetic data have been proposed in the literature, however, these approaches typically require more hand-labeled data (or language-specific knowledge). In particular, the *tagged corruption method* (Stahlberg and Kumar, 2024) mines likely confusions from gold data that are used to corrupt monolingual data. This approach relies on gold training data for mining the error patterns and a tagger model that can tag the errors. This approach performed well on English, but did not perform well in a multilingual setting (Stahlberg and Kumar, 2024). Notably, the results on Russian (RULEC) for a similar model to ours are not competitive with the results we obtain in this work (see Section 6.1).

Several works employ round-trip machine translation where full-sentence translations are used (Lichtarge et al., 2019; Kementchedjhieva and Søgaard., 2023; Zhou et al., 2020). We compare with this approach in Section 6.1.

Xie et al. (2018) and Kiyono et al. (2019) use *back-translation*, where a machine translation model is trained in reverse direction (from well-formed to ungrammatical sentences). The approach is applied to English GEC, where over 1M gold sentence pairs are used to train the translation model for English. In contrast, we only have 5K and 32K gold sentence pairs for Russian and Ukrainian, respectively (please also see our further discussion in the Limitations).

4.2 Our Approach (SeLex-RT)

In the SeLex-RT approach, we generate *confusion sets* by translating a native corpus T in language l (e.g. Russian) into another language (*pivot*) and then back into l. We token-align the original sentence with its back-translated version using the alignment model of Sabet et al. (2020). A target word t that occurs in T in a diverse set of contexts, is expected to have different back-translations determined by each unique context. A confusion set for t will include all unique back-translated words aligned to t, for all occurrences of t in T. Our expectation is that because translations are imperfect, the back-translated sentences will not be identical to the original sentence. Furthermore, because word sense nuances can be challenging for non-native speakers due to the differences in contextual word usage of ambiguous words in their native language and in the second language, on the lexical level, the differences in back-translation will be similar to mistakes made by language learners. The intuition that this approach of round-trip translation with token-based alignments will generate errors similar to learner mistakes is based on a recent study that used this method for generating vocabulary exercises for English language learners (Panda et al., 2022; Palma Gomez et al., 2023). Indeed, we show below that the method is able to replicate well real learner mistakes.

We stress that, in contrast to the full-sentence translation approach (Section 6.1), we only extract aligned word pairs used to corrupt the data. Furthermore, we use multiple translation hypotheses, thereby generating multiple back-translations for each token in a single sentence.

4.3 Generating Synthetic Errors with SeLex-RT and the Baseline Methods

We now compare the *SeLex-RT* method and the baseline methods *Spell* and *Morph* with respect to their ability to generate errors that mimic real learner errors.[4] The three methods all produce *token-level confusion sets* that are used to corrupt monolingual data. Examples of confusion sets (in English) for the word *walk* are {walk,talk,wall}, {walk,walking,walked}, and {walk,go,stroll}, generated with the Spell, Morph, and SeLex-RT method, respectively. If *walk* occurs in a sentence and is selected to be corrupted, it is replaced with another candidate from its corresponding confusion set (generated with the corresponding corruption method). Appendix Table C4 illustrates sample confusion sets in Russian.

We use these methods to generate the erroneous side of a sentence from a monolingual corpus, by introducing errors into it. We iterate over the tokens in the sentence and select tokens to be replaced with another token from the confusion set of that token.[5] An example of a sentence (in Russian) produced with each synthetic data generation method can be found in Appendix Table C5.

The errors are generated in a 15M monolingual Russian corpus (Sorokin, 2017), used to train GEC models (see next Section). SeLex-RT confusion sets are generated, by taking a sample of 100K sentences from the 15M corpus. The 100K sentence sample is translated into English and then back into Russian.[6] We use the translation systems of Tiedemann and

[4] We do not include the *Char* method here, as it only performs character-level perturbations.

[5] The tokens to be replaced are selected with prob. 15%.

[6] We only use 100K sentences to generate confusion sets, which are then used to corrupt the entire 15M corpus.

Synth. method	RULEC		RU-Lang8	
	Gram.	Lex.	Gram.	Lex.
Spell	41.9	12.0	34.4	9.8
Morph	**75.9**	5.0	**72.4**	5.6
SeLex-RT	58.8	**46.7**	65.1	**51.5**

Table 3: Percentage of unique errors (represented as source-target pairs) in Russian learner test data that have been produced with each synthetic method. Results broken down by error group. Best result is in bold. Confusion set size is determined based on the average number of corruption options for a target token.

Thottingal (2020), with English as the pivot. We choose English as there are typically high-quality translation systems available to and from English for a variety of languages (but also see Limitations). For each sentence, 10 forward translations, and 10 backward translations for each forward translation are produced, for a total of 100 back-translations per sentence. In 6.1, we discuss the computational costs of the approach.

How well do synthetic errors created with each method replicate learner errors? To answer this question, we compute the percentage of unique errors (represented as source-target pairs) in the Russian learner datasets that are found in the synthetic data produced with each method (*target* refers to the correct word, and *source* refers to the potentially erroneous word in learner data or the replacement used to corrupt the data with a synthetic method). Table 3 demonstrates that the SeLex-RT method generates about half of learner errors in the lexico-morph. category, while the baseline methods produce only a small fraction, confirming that the SeLex-RT-based confusion sets are able to better mimic real learner errors, compared to the other methods.

5 Experimental Setup

Model architecture Drawing on the methods that showed superior performance in multilingual GEC (Rothe et al., 2021; Palma Gomez et al., 2023), we chose the seq2seq framework (see Section 2): We implement two types of Transformer seq2seq models – a smaller one trained from scratch, and a larger model that uses mT5 as a starting point.

Model 1 Our (smaller) model is pre-trained on synthetic data and finetuned on the gold data. It has 275M parameters.

Model 2 We adopt the approach of Rothe et al. (2021), making use of mT5 (Xue et al., 2021). Rothe et al. (2021) finetune mT5 on GEC gold data only, but Palma Gomez et al. (2023) show the importance of a 2-step finetuning, which we adopt here: in the first step, mT5 is finetuned on the synthetic data, and in the second step we finetune on the gold training data. Due to computational constraints, we use mT5-Base that has 580M parameters. Larger models (large, xl, xxl) lead to stronger results both with 1-step finetuning on gold data only (Rothe et al., 2021) and with a 2-step finetuning on Russian (Palma Gomez and Rozovskaya, 2024) and Ukrainian (Palma Gomez et al., 2023). Appendix A lists the training details and the hyperparameters.

Baseline synthetic methods We implement four data generation techniques: (1) character-level transformations (*Char*); (2) spell-based transformations (*Spell*); and (3) morphology-based transformations (*Morph*), (4) and a combination *Morph+Spell*.[7] Appendix C provides more detail on how we generate errors for the two languages.

Synthetic data The synthetic data is created by corrupting *monolingual* data from the Yandex corpus for Russian (Sorokin, 2017), and from the CC-100 dataset for Ukrainian (Wenzek et al., 2020). Each training example in the synthetic data consists of a pair of sentences: a corrupted sentence and its correct counterpart from the monolingual corpus. Example is shown in Appendix Table C5.

Evaluation To compare with published work, we adopt the MaxMatch scorer (Dahlmeier and Ng, 2012) for Russian, and ERRANT for Ukrainian (Bryant et al., 2017). All results are on the test partitions. In Russian, we apply a spellchecker (Rozovskaya, 2022) to the data as pre-processing.

6 Key Results

Baselines We use the smaller Model 1 to compare the baseline methods. Model 1 is trained on 15M synthetic sentence pairs, while the mT5-models are trained on 2M synthetic sentence pairs (due to computational constraints). Gold training data (in Table 1) is used to finetune the models.

Key results on Russian and Ukrainian Key results on Russian and Ukrainian are shown in Tables 4 and 5, respectively. Detailed performance that includes precision and recall is shown in Appendix Tables D6 and D7.

As the tables show, the *Char* method is the weakest baseline, whereas *Morph* outperforms *Spell*

[7] In Morph+Spell, the Morph method is applied to each sentence first, followed by Spell.

Synth. method	$F_{0.5}$	
	RULEC	RU-Lang8
Model 1 (Baselines)		
Char	40.0	40.2
Spell	54.0	52.6
Morph	57.5	56.0
M+S	**60.7**	**61.6**
SeLex-RT	49.1	47.7
Model 1 (Best baselines with SeLex-RT)		
M+SeLex-RT	**60.7***	**59.9***
M+S+SeLex-RT	**62.1***	**62.6***
Model 2		
M+S	62.0	59.9
M+S+SeLex-RT	**62.9***	**60.9***

Table 4: Key results on Russian. Best $F_{0.5}$ for each dataset and model are in bold. *Model 1* denotes models trained from scratch, while *Model 2* refers to models that finetune mT5-Base. *M+S* stands for *Morph+Spell.* Results marked with a ⋆ are statistically significant at the .03 level, compared to the respective baseline.

Synth. method	$F_{0.5}$
Model 1 (Baselines)	
Char	55.2
Spell	62.8
Morph	43.8
M+S	**63.3**
SeLex-RT	57.6
Model 1 (Best baselines+SeLex-RT)	
Spell+SeLex-RT	**64.7**
M+S+SeLex-RT	64.3
Model 2	
Spell	64.1
Spell+SeLex-RT	**66.1**

Table 5: Key results on the Ukrainian benchmark UA-GEC. Best $F_{0.5}$ for each model is in bold. *Model 1* denotes models trained from scratch, while *Model 2* refers to models that finetune mT5. *M+S* stands for *Morph+Spell.*

for Russian, while *Spell* outperforms *Morph* for Ukrainian. The combination *Morph+Spell* produces a stronger model for both languages. We also show the SeLex-RT performance on its own. The middle part of the tables shows the results of adding SeLex-RT to the best baselines. SeLex-RT errors are added to the text after adding errors produced with the baseline methods.[8] The bottom part of the tables compares Model 2 types using the best baseline data generation strategy.

Findings *The improvements from adding SeLex-RT are consistent across all models and both languages.* The differences between the best models with the SeLex-RT component and their respective baselines are all statistically significant at the .03 level on both Russian benchmarks.[9] We do not perform statistical significance testing on UA-GEC since the gold references are not publicly available. On Russian, improvements are larger for Model 1 than for Model 2. By contrast, improvements in Ukrainian are slightly larger for Model 2. We believe this may signal that using a pre-trained language model (mT5) is more beneficial for Russian, as the mC4 dataset on which mT5 is pre-trained contains 18 times less Ukrainian data compared to Russian, thus the SeLex-RT component could have a larger impact on Ukrainian. This finding is promising for low-resource GEC scenarios.

Comparing the two model types, Model 2 outperforms Model 1, with the exception for RU-Lang8. We hypothesize this may be due to overfitting (we are finetuning on RULEC data, which is really small), and thus a model with more parameters is more likely to overfit, resulting in a less optimal performance on out-of-domain RU-Lang8 data.

6.1 Comparison to Previous Work

Comparison to other translation methods In Table 6 we compare SeLex-RT with the round-trip machine translation method (henceforth MT) in Lichtarge et al. (2019) that uses full-sentence translations to generate the source side of the parallel data (detailed performance with precision and recall scores is shown in Appendix Table D8). We use English as the pivot and translate the 15M target side of the monolingual data. The round-trip translations are generated using the MT systems of Tiedemann and Thottingal (2020), the same system used to generate round-trip translations for SeLex-RT. The difference is that with SeLex-RT we only use confusion sets obtained from the round-trip translations, whereas for the round-trip MT method, following Lichtarge et al. (2019), we use full-sentence round-trip translations as synthetic training sentence pairs.

The original round-trip MT method (Lichtarge et al., 2019) (listed as *MT only (orig)* in the table) exhibits low precision. Upon further inspection, we conclude that this is due to the MT method introduc-

[8] Note that when combining Spell and Morph, Morph and SeLex-RT, and Morph with Spell and SeLex-RT, the same number of training sentences is used in all cases. SeLex-RT errors are generated with probability 0.1, set experimentally.

[9] We use a two-sided approximate randomization test (Graham et al., 2014). We used the implementation of Alhafni et al. (2023).

Synth. method	$F_{0.5}$ RULEC	$F_{0.5}$ RU-Lang8
3 configurations of the round-trip MT method		
MT only (orig)	40.1	42.9
Morph+MT	56.1	55.7
M+S+MT	57.8	60.7
This work		
M+S+SeLex-RT	**62.1***	**62.6***

Table 6: Comparison with full-sentence round-trip translation method in Lichtarge et al. (2019). Results marked with a ⋆ are statistically significant at the .05 level, compared to the best MT configuration above.

ing many changes (including low-quality changes). To address the issue, we combine MT with our two strongest baselines – Morph and Morph+Spell, as follows: (1) for half of the training sentences, we apply the baseline methods on top of the full-sentence round-trip MT translation, while (2) for the other 50% of the data, we apply the baseline synthetic method only, without the round-trip MT. 1 or 2 is chosen uniformly at random for each sentence. The resulting models are shown as Morph+MT and M+S+MT in Table 6.[10] These combinations improve precision, but hurt recall. Our conclusion is thus that *our method (SeLex-RT) introduces more high-quality changes*, which yield higher recall and do not hurt precision. As the table shows, our approach substantially outperforms all the models that use full-sentence round-trip MT.[11]

Computational costs of SeLex-RT: MT systems are known to incur high computational costs. It is important to emphasize that with SeLex-RT (in contrast to Lichtarge et al. (2019) we do not translate the entire synthetic training set, but only a small subset – 100K sentences. To elaborate further: using 10 translations in each direction would amount to 10M translations, which is less expensive than translating 15M sentences used to train Model 1. Further, as we show in 7.3, using even 5 translations in each direction (for a total of 2.5M translations) already results in substantial improvements that are almost as good as using 10 translations. Finally, with SeLex-RT confusions, we can generate an unlimited number of synthetic sentence pairs.

Model	$F_{0.5}$ RULEC	$F_{0.5}$ RU-Lang8
P&R'24 (Model 1)	57.6	56.0
This work (Model 1)	62.1	62.6
P&R'24	64.8	62.1
This work (Model 2)	62.9	60.9

Table 7: Comparison with previous work for RULEC and RU-Lang8 using enriched references. P&R'24 is mT5-Large trained on 10M sentence pairs in Palma Gomez and Rozovskaya (2024). *Model 1* is pre-trained on 15M sentence pairs. *Model 2* denotes mT5-Base pre-trained on 2M synthetic sentence pairs. Both models use Morph+Spell+SeLex-RT errors.

Model	$F_{0.5}$ RULEC	$F_{0.5}$ RU-Lang8
This work (M+S+SeLex-RT)		
Model 1 (15M)	49.2	51.8
Model 2 (2M)	51.2	52.3
Previous work (similar model arch. to ours)		
P&R (15M)	47.4	47.7
P&R mT5-B (10M)	51.0	49.8
P&R mT5-L (10M)	53.2	54.5
S&K mT5-B (2.5M)	26.4	-
S&K mT5-XXL (2.5M)	44.3	-

Table 8: Comparison with previous work for Russian, using original references. The top segment shows models trained in this work. The second segment shows previous work with similar model architectures to ours. *S&K* stands for Stahlberg and Kumar (2024). P&R stands for Palma Gomez and Rozovskaya (2024). mT5-B and mT5-L stand for mT5-Base and mT5-Large, respectively. In the parentheses, we show the number of synthetic training examples.

Comparison to other GEC methods We compare to the results in Palma Gomez and Rozovskaya (2024) with enriched Russian benchmarks (Table 7). They implement similar architectures to our Model 1 and 2, but use *Morph* transformations. Their Model 2 is mT5-Large pre-trained on 10M sentence pairs, whereas we use mT5-Base and pre-train on 2M sentence pairs. On Model 1, we outperform Palma Gomez and Rozovskaya (2024) by 5 points on RULEC and 7 points on RU-Lang8. On Model 2, we are 2 points below their results, however, since we use similar architectures, we expect that repeating our experiments with mT5-Large will at least match their results.

Comparison on Russian using original references We compare with prior work using the orig-

[10] We tried several ways of combining MT with Morph and Spell (with different probabilities) and show the best ones.

[11] We note that the SeLex-RT candidates also contain noise, filtered naturally in the corruption stage: a replacement candidate is chosen based on the relative frequency of their alignment (i.e. how many times a round-trip translation was observed with the specific target token in our data). This way, noisy candidates that typically occur less frequently compared to the high-quality translations, are chosen less frequently.

Model	**F$_{0.5}$**
This work (Model 1 best, 15M)	64.7
This work (Model 2 best, 2M)	66.1
P. et al.(Model 1, 35M)	66.0
P. et al. (mT5-Large, 10M)	68.1
Bondarenko et al. (2023)	68.2

Table 9: Comparison with previous work for Ukrainian. P. et al denotes Palma Gomez et al. (2023).

inal single-reference annotations in Table 8. The top segment of the table shows models trained in this work. The second segment shows models that are directly comparable to ours due to similar architectures and/or model parameters. We outperform all models in segment 2 (with the exception of a much more powerful mT5-Large pre-trained on 10M sentence pairs), often by a large margin, even when we use less synthetic data. Note that S&K is a recent multilingual implementation of tagged corruption models (Stahlberg and Kumar, 2024) with the same architecture and similar amounts of synthetic data used to ours, but only obtains a result of 26.4 compared to our result of 51.2 on RULEC. This suggests the importance of high-quality synthetic data even when using pre-trained models in GEC. Appendix D.3 provides an expanded version of Table 8, where we compare a variety of models from previous works. We show that our models are competitive with or better than state-of-the-art.

Comparison to previous work on Ukrainian This is shown in Table 9. SOTA on Ukrainian use larger models and more data. Our best result of 66.1 is 2 points below the 2 systems that placed first in the shared task. However, these models use more knowledge and data (Bondarenko et al., 2023; Palma Gomez et al., 2023; Syvokon and Romanyshyn, 2023). Palma Gomez et al. (2023) is the same architecture as our Model 2 but uses mT5-Large and 10M synthetic sentence pairs.

7 Additional Analyses on Russian

7.1 SeLex-RT and Performance by Error Type

Since our focus is on lexical errors, we evaluate performance by coarse error type for Russian. To this end, both gold errors and errors flagged by the systems are classified using an ERRANT-style tool for Russian (Rozovskaya, 2022) (see Table 2). Results ($F_{0.5}$) on RU-Lang8 (for grammar and lexico-morph. errors) are shown in Table 10.

	F$_{0.5}$	
Synth. method	**Gram.**	**Lex., morph.**
Model 1 (Baselines)		
Spell	56.4	13.8
Morph	60.5	6.9
M+S	69.8	15.4
SeLex-RT	51.2	20.3
Model 1 (M+S+SeLex-RT)		
M+S+SeLex-RT	**71.1**	**29.9**
Model 2		
M+S	67.2	22.8
M+S+SeLex-RT	**69.4**	**27.0**

Table 10: Results by coarse error group on RU-Lang8.

		Performance (F$_{0.5}$)		
Synth. method	**Ref. set**	**Gram.**	**Lex., morph.**	**Total**
Model 1 (Baselines)				
Spell	RG	63.3	13.3	60.8
	CG	70.5	16.1	71.6
Morph	RG	59.1	12.5	60.8
	CG	74.6	30.7	75.3
M+S	RG	66.7	22.3	65.0
	CG	78.1	33.7	78.3
Model 1 (M+S+SeLex-RT)				
	RG	66.8	32.2	64.3
	CG	**79.6**	**59.6**	**79.7**

Table 11: Evaluation with three standard references (RG) and closest golds (CGs) on a 500-sentence subset (Model 1) on RULEC. Best results are in bold.

Adding SeLex-RT improves performance on lexico-morph. errors. The performance on grammar errors improves as well, indicating that the SeLex-RT errors complement the baseline methods in that category. Tables D10 and D9 in Appendix show results on 4 coarse categories on both Russian benchmarks, revealing similar trends on RULEC.

7.2 Evaluation with Closest Gold References

Evaluation in GEC is known to be notoriously difficult (Choshen and Abend, 2018b; Bryant and Ng, 2015; Felice and Briscoe, 2015; Napoles et al., 2015). This is because the space of valid corrections for a given sentence is large (Choshen and Abend, 2018b,a), but the set of human references in evaluation is limited. The prevalent use of too few references is known to underestimate model performance, especially on lexical errors (Choshen and Abend, 2018b), which explains low precision on these errors in Section 7.1.

To evaluate performance more accurately, we generate *closest gold* (CGs) references, i.e. *cor-*

Hypo pool size	Performance ($F_{0.5}$)					
	RULEC			RU-Lang8		
	P	R	$F_{0.5}$	P	R	$F_{0.5}$
	Baseline (Morph+Spell)					
NA	68.8	41.2	60.7	69.5	41.4	61.2
	Morph+Spell+SeLex-RT					
1	69.9	40.9	61.2	69.3	42.6	61.6
25	69.7	42.7	61.9	69.6	45.0	62.7
100	71.0	42.0	**62.3**	69.9	44.2	62.6
256	70.5	42.6	**62.3**	70.0	44.7	**62.9**

Table 12: Size of translation pool. Evaluation on Russian using 3 references with smaller models (Model 1 type). Best results are in bold.

rected versions constructed relative to the system output instead of the source sentence, akin to post-evaluation for machine translation (Rozovskaya and Roth, 2021). Due to the cost of generating CGs (a separate set of references needs to be generated for each system output), we perform this evaluation for a subset of 500 sentences for Model 1. Appendix D.2 provides detail on the annotation.

Results Table 11 shows evaluation on RULEC. Appendix Tables D12 and D13 show detailed results on both datasets. The key findings are: (1) Precision on lexico-morph. errors is as high as on grammar errors, confirming that, indeed, *performance is more severely underestimated on lexical errors*; (2) The SeLex-RT component improves recall on lexical errors by more than 20 points, whereas precision numbers remain just as high. Overall, evaluation with CGs demonstrates that *the SeLex-RT component significantly improves performance on lexical errors*, compared to the baselines.

7.3 Varying the Number of Translations

Thus far, we have used 100 round-trip translations per sentence, to generate confusion candidates. We refer to the 100 translations as *the hypothesis pool*. To evaluate the effect of the pool size on performance, we vary the number of hypotheses in each direction, and use 1, 5, 10, and 16, for a total number of translations for a sentence being 1, 25, 100, and 256. The average confusion set size increases from 3 to 120, as the pool increases from 1 to 256.

Table 12 shows evaluation with different pool sizes. While every setting outperforms the baseline, there is no significant difference in performance, as the translation pool is expanded beyond 25 hypotheses, suggesting that going further down the list of back-translations is not beneficial, perhaps due to the degrading quality of the translations.

1.	значит?/означает	*means?/signifies*
2.	следовать?/соблюдать	follow?/observe
3.	обучения?/знаний	learning?/knowledge
4.	норм?/правил	norms?/rules
5.	несколько?/некоторые	some?/several
6.	есть?/находятся	exist?/are

Table 13: Examples of lexical errors (in Russian, with the English translations) missed by models trained on the synthetic data generated methods but corrected by models that inlcude the SeLex-RT component.

7.4 Error Analysis

We analyze model output and identify interesting examples of errors corrected by the model with the SeLex-RT component (these errors are missed by the models trained on standard synthetic data). Table 13 illustrates some of the errors. In Appendix Table D15 we show these examples within sentences. Observe that the model corrects mistakes that do not share the root/stem (examples 2, 3, 6), as well as those that share the root (example 1) but have diverging derivations. Example 5 illustrates an interesting and challenging learner error in the use of indefinite pronouns. Acquisition of subtle semantic nuances of indefinite pronouns pose a challenge to English learners (Rabinovich et al., 2019), and we also observe this in Russian data.

8 Conclusion

We present a novel approach to creating synthetic data for correcting language learner lexical errors. We hypothesized, drawing on evidence from second language acquisition research, that challenges facing language learners will also manifest in the machine translation output, as many lexical errors result from deviations in the second language from the native usage. Our approach creates *controlled* confusions from token-aligned sentences and their translations. Crucially, we do not use full-sentence back-translation and generate multiple translation hypotheses, which allows us to create diverse and high-quality synthetic errors. Extensive experiments on two low-resource languages – Russian and Ukrainian – and three benchmarks, and additional analyses on Russian demonstrate the effectiveness of our approach. Future work will focus on the application of this method to other languages.

Limitations

Availability of MT systems for low-resource languages One limitation of our approach is that it requires the use of machine translation systems to translate the target language data into a pivot language and back. While for many languages, good MT systems are available today (typically to and from English), for some low-resource languages, MT systems are not available or are extremely weak. Our expectation is that the method will require a relatively good MT system in both directions (forward and backward). In fact, as we use 256 round-trip translations for Russian, we do not observe the performance improvement (see Section 7.3), suggesting that the degrading quality of translations affect the quality of generated confusion sets. It is worth noting that we only used a single pivot – English – due to the fact that there are typically high-quality translation systems available to and from English for a variety of languages.

Analyzing the impact of the translation systems employed In this work, we have used one pivot language (English) and a single MT system for each target language. Using multiple pivot languages is outside the scope of the paper, but we hope to extend this work in the future and investigate this interesting question. It should be noted that this experiment might conflate the impact of a pivot with the quality of an MT system.

Another question concerns the quality of an MT system used when using the same pivot. While we only used one type of MT system for each language, our experiments with hypothesis pool in Section 7.3 show that going down the list of translations is not beneficial (beyond 10 top-scoring translations) is not beneficial. We hypothesize that this indicates that higher-quality translation at the top of the hypothesis list are preferable for generating confusion sets for our task.

Coverage of diverse error types We focus on single-word lexical errors, and do not address other multi-token replacements and less-preferred word order. Extending the approach to multi-token replacements is planned for future work. We believe that the SeLex-RT approach can be fit to address these types of errors, as well.

It should also be pointed out that SeLex-RT operates on the assumption of correcting lexical errors resulting from first language interference and the similarities between learner and translation errors. However, in terms of first language backgrounds, while in RULEC, all learners are native English speakers, in RU-Lang8 and in the UA-GEC dataset there is a variety of first language backgrounds. We have not explored the potential of utilizing various pivot languages to fit the language background of the learners, and we leave it for future work.

Approaches appropriate for low-resource GEC There are several definitions on low-resource settings in natural language processing. In this work, we follow the definition of low-resource as languages that only have a small amount of hand-labeled data available. However, the proposed approach, SeLex-RT, assumes the availability of high-quality machine translation systems, as well as resources required to generate synthetic errors (a spellchecker and a part-of-speech tagger). We also acknowledge the limitations of our experiments, specifically, the Morph baseline that we have implemented only uses morphological transformations, without the patterns, as in (Choe et al., 2019). One reason for this is that their approach was found effective for correcting grammar errors but not for lexical errors (White and Rozovskaya, 2020). We further hypothesize that the amount of hand-labeled data available to us is not sufficient to extract high-quality patterns, however, we leave this as future work to investigate whether adding patterns mined from hand-labeled (training or development) data is effective for correcting lexical errors.

We have selected several baselines that we consider to be appropriate for multilingual settings that require several resources, but do not require large amounts of hand-labeled data. We discuss several other methods in Section 4.1 to be more resource- or knowledge-intensive, such as the back-translation approach (Xie et al., 2018). That said, recent work on low-resource machine translation (Tan and Zhu, 2024) can train models with under 50K training examples, which is still significantly more than what is available for languages other than English as hand-labeled data. Nevertheless, we believe this is an intriguing research question that could be explored in the context of GEC, but is orthogonal to the solution we propose.

Use of reference-based evaluations for performance on lexical errors Finally, we also note that reference-based evaluation is challenging when lexical errors are concerned, and evaluation with closest golds (CGs) is expensive and not feasible on a large scale. This problem arises in several recent works that attempted to gauge the performance of large language models (LLMs) on GEC, including ChatGPT (Fang et al., 2023; Katinskaia and Yangarber, 2024). We believe this is the same issue we encountered in this work when trying to evaluate the performance on lexical mistakes. We leave this as a direction for future work.

Acknowledgments

The authors thank the annotators for their annotation work. The authors are grateful to the anonymous ARR reviewers for their insightful comments.

References

Bashar Alhafni, Go Inoue, Christian Khairallah, and Nizar Habash. 2023. Advancements in Arabic grammatical error detection and correction: An empirical investigation. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 6430–6448, Singapore. Association for Computational Linguistics.

Anna Alsufieva, Olsya Yatsenko Kisselev, and Sandra G. Freels. 2012. Results 2012: Using flagship data to develop a russian learner corpus of academic writing. *Russian Language Journal*, 62:79–105.

Abhijeet Awasthi, Sunita Sarawagi, Rasna Goyal, Sabyasachi Ghosh, and Vihari Piratla. 2019. Parallel iterative edit models for local sequence transduction. 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)*. Association for Computational Linguistics.

Maksym Bondarenko, Artem Yushko, Andrii Shportko, and Andrii Fedorych. 2023. Comparative study of models trained on synthetic data for Ukrainian grammatical error correction. In *Proceedings of the Second Ukrainian Natural Language Processing Workshop (UNLP)*. Association for Computational Linguistics.

Andrey Bout, Alexander Podolskiy, Sergey Nikolenko, and Irina Piontkovskaya. 2023. Efficient grammatical error correction via multi-task training and optimized training schedule. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Christopher Bryant, Mariano Felice, Øistein E. Andersen, and Ted Briscoe. 2019. The BEA-19 shared task on grammatical error correction. In *Proceedings of the ACL Workshop on Innovative Use of NLP for Building Educational Applications (BEA-19)*.

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)*. Association for Computational Linguistics.

Christopher Bryant and Hwee Tou Ng. 2015. How far are we from fully automatic high quality grammatical error correction? In *ACL*.

Christopher Bryant, Zheng Yuan, Muhammad Reza Qorib, Hannan Cao, Hwee Tou Ng, and Ted Briscoe. 2023. Grammatical error correction: A survey of the state of the art. *Computational Linguistics*, pages 643–701.

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*. 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 AAAI*. Association for the Advancement of Artificial Intelligence.

Leshem Choshen and Omri Abend. 2018a. Automatic metric validation for grammatical error correction. In *ACL*.

Leshem Choshen and Omri Abend. 2018b. Inherent biases in reference-based evaluation for grammatical error correction and text simplification. In *ACL*.

Daniel Dahlmeier and Hwee Tou Ng. 2011. Grammatical error correction with alternating structure optimization. In *Proceedings of ACL*.

Daniel Dahlmeier and Hwee Tou Ng. 2012. Better evaluation for grammatical error correction. In *Proceedings of NAACL*.

Ali Derakhshan and Elham Karimi. 2015. The interference of first language and second language acquisition. *Theory and Practice in Language Studies*, 5(10):2112–2117.

Tao Fang, Shu Yang, Kaixin Lan, Derek F. Wong, Jinpeng Hu, Lidia S. Chao, and Yue Zhang. 2023. Is ChatGPT a highly fluent grammatical error correction system? In *arXiv preprint arXiv:2304.01746*.

Mariano Felice and Ted Briscoe. 2015. Towards a standard evaluation method for grammatical error detection and correction. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Association for Computational Linguistics.

Simon Flachs, Felix Stahlberg, and Shankar Kumar. 2021. Data strategies for low-resource grammatical error correction. In *Proceedings of the 16th Workshop on Innovative Use of NLP for Building Educational Applications*, pages 117–122, Online. Association for Computational Linguistics.

Yvette Graham, Nitika Mathur, and Timothy Baldwin. 2014. Randomized significance tests in machine translation. In *Proceedings of the Ninth Workshop on Statistical Machine Translation*. Association for Computational Linguistics.

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*.

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 ACL Workshop on Innovative Use of NLP for Building Educational Applications (BEA-19)*.

Jianshu Ji, Qinlong Wang, Kristina Toutanova, Yongen Gong, Steven Truong, , and Jianfeng Gao. 2017. A nested attention neural hybrid model for grammatical error correction. In *ACL*.

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.

Anisia Katinskaia and Roman Yangarber. 2024. GPT-3.5 for grammatical error correction. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*. ELRA and ICCL.

Satoru Katsumata and Mamoru Komachi. 2019. (almost) unsupervised grammatical error correction using synthetic comparable corpus. In *Proceedings of the ACL Workshop on Innovative Use of NLP for Building Educational Applications (BEA-19)*.

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*.

Yova Kementchedjhieva and Anders Søgaard. 2023. Grammatical error correction through round-trip machine translation. In *Findings of the Association for Computational Linguistics: EACL 2023*. 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.

Shaopeng Lai, Qingyu Zhou, Jiali Zeng, Zhongli Li, Chao Li, Yunbo Cao, and Jinsong Su. 2022. Type-driven multi-turn corrections for grammatical error correction. In *Findings of the Association for Computational Linguistics: ACL 2022*, Dublin, Ireland. Association for Computational Linguistics.

Claudia Leacock, Martin Chodorow, Michael Gamon, and Joel Tetreault. 2010. *Automated Grammatical Error Detection for Language Learners*. Morgan and Claypool Publishers.

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.

Eric Malmi, Sebastian Krause, Sascha Rothe, Daniil Mirylenka, and Aliaksei Severyn. 2019. Encode, tag, realize: High-precision text editing. In *EMNLP-IJCNLP*.

Masato Mita, Tomoya Mizumoto, Masahiro Kaneko, Ryo Nagata, and Kentaro Inui. 2019. Cross-corpora evaluation and analysis of grammatical error correction models – is single-corpus evaluation enough? In *NAACL*.

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.

Courtney Napoles, Keisuke Sakaguchi, Matt Post, and Joel Tetreault. 2015. Ground truth for grammatical error correction metrics. In *ACLIJCNLP*.

Courtney Napoles, Keisuke Sakaguchi, and Joel Tetreault. 2017. JFLEG: A fluency corpus and benchmark for grammatical error correction. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 229–234, Valencia, Spain. Association for Computational Linguistics.

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 CoNLL: Shared Task*.

Kostiantyn Omelianchuk, Vitaliy Atrasevych, Artem Chernodub, and Oleksandr Skurzhanskyi. 2020. GECToR ? Grammatical Error Correction: Tag, Not Rewrite . In *Building Educational Applications Workshop (BEA)*.

Kostiantyn Omelianchuk, Andrii Liubonko, Oleksandr Skurzhanskyi, Artem Chernodub, Oleksandr Korniienko, and Igor Samokhin. 2024. Pillars of grammatical error correction: Comprehensive inspection of contemporary approaches in the era of large language models. In *Proceedings of the 19th Workshop* *on Innovative Use of NLP for Building Educational Applications (BEA 2024)*. Association for Computational Linguistics.

Frank Palma Gomez, Subhadarshi Panda, Michael Flor, and Alla Rozovskaya. 2023. Using neural machine translation for generating diverse challenging exercises for language learner. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics.

Frank Palma Gomez and Alla Rozovskaya. 2024. Multi-reference benchmarks for Russian grammatical error correction. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1253–1270, St. Julian's, Malta. Association for Computational Linguistics.

Frank Palma Gomez, Alla Rozovskaya, and Dan Roth. 2023. A low-resource approach to the grammatical error correction of ukrainian. In *Proceedings of the Second Ukrainian Natural Language Processing Workshop in conjunction with EACL*.

Subhadarshi Panda, Frank Palma Gomez, Michael Flor, and Alla Rozovskaya. 2022. Automatic generation of distractors for fill-in-the-blank exercises with round-trip neural machine translation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop*. Association for Computational Linguistics.

Muhammad Reza Qorib, Alham Fikri Aji, and Hwee Tou Ng. 2024. Efficient and interpretable grammatical error correction with mixture of experts. In *Findings of the Association for Computational Linguistics: EMNLP 2024*. Association for Computational Linguistics.

Muhammad Reza Qorib and Hwee Tou Ng. 2023. System combination via quality estimation for grammatical error correction. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Ella Rabinovich, Julia Watson, Barend Beekhuizen, and Suzanne Stevenson. 2019. Say anything: Automatic semantic infelicity detection in L2 english indefinite pronouns. In *Proceedings CoNLL*.

Sascha Rothe, Jonathan Mallinson, Eric Malmi, Sebastian Krause, and Aliaksei Severyn. 2021. A simple recipe for multilingual grammatical error correction. In *ACL*.

Alla Rozovskaya. 2022. Automatic Classification of Russian Learner Errors. In *LREC*.

Alla Rozovskaya and Dan Roth. 2019. Grammar error correction in morphologically-rich languages: The case of russian. In *Transactions of ACL*.

Alla Rozovskaya and Dan Roth. 2021. How good (really) are grammatical error correction systems? In *EACL*.

Alla Rozovskaya, Dan Roth, and Mark Sammons. 2017. Adapting to learner errors with minimal supervision. *Computational Linguistics. To appear.*

Masoud Jalili Sabet, Philipp Dufter, and Hinrich Schütze. 2020. Simalign: High quality word alignments without parallel training data using static and contextualized embeddings. *arXiv preprint arXiv:2004.08728*.

Alexey Sorokin. 2017. Spelling correction for morphologically rich language: a case study of russian. In *Proceedings of the 6th Workshop on Balto-Slavic Natural Language Processing*.

Alexey Sorokin. 2022. Improved grammatical error correction by ranking elementary edits. In *EMNLP*.

Alexey Sorokin, Alexey Baytin, Irina Galinskaya, Elena Rykunova, and Tatiana Shavrina. 2016. SpellRuEval: the first competition on automatic spelling correction for russian. In *Proceedings of the International Conference "Dialogue 2016"*.

Felix Stahlberg, Christopher Bryant, and Bill Byrne. 2019. Neural grammatical error correction with finite state transducers . In *NAACL*.

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)*. Association for Computational Linguistics.

Felix Stahlberg and Shankar Kumar. 2024. Synthetic data generation for low-resource grammatical error correction with tagged corruption models. In *Proceedings of the 19th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2024)*, Mexico City, Mexico. Association for Computational Linguistics.

Xin Sun and Houfeng Wang. 2022. Adjusting the precision-recall trade-off with align-and-predict decoding for grammatical error correction. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*. Association for Computational Linguistics.

Oleksiy Syvokon and Mariana Romanyshyn. 2023. The UNLP 2023 shared task on grammatical error correction for Ukrainian. In *Proceedings of the Second Ukrainian Natural Language Processing Workshop in conjunction with EACL*.

William Tan and Kevin Zhu. 2024. Nusamt-7b: Machine translation for low-resource indonesian languages with large language models.

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)*.

Jörg Tiedemann and Santhosh Thottingal. 2020. OPUS-MT — Building open translation services for the World. In *Proceedings of the 22nd Annual Conferenec of the European Association for Machine Translation (EAMT)*, Lisbon, Portugal.

Viet Anh Trinh and Alla Rozovskaya. 2021. New dataset and strong baselines for the grammatical error correction of russian. In *ACL Findings*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, Advances in Neural Information Processing Systems*.

Guillaume Wenzek, Marie-Anne Lachaux, Alexis Conneau, Vishrav Chaudhary, Francisco Guzmán, Armand Joulin, and Edouard Grave. 2020. CCNet: Extracting high quality monolingual datasets from web crawl data. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*.

Maxwell White and Alla Rozovskaya. 2020. A comparative study of synthetic data generation methods for grammatical error correction. In *BEA*.

Ziang Xie, Guillaume Genthial, Stanley Xie, Andrew Ng, and Dan Jurafsky. 2018. Noising and denoising natural language: Diverse backtranslation for grammar correction. 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 619–628, New Orleans, Louisiana. Association for Computational Linguistics.

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 *NAACL*.

Michihiro Yasunaga, Jure Leskovec, and Percy Liang. 2021. LM-Critic: Language Models for Unsupervised Grammatical Error Correction . In *EMNLP*.

Jingheng Ye, Yinghui Li, Yangning Li, and Hai-Tao Zheng. 2023. Mixedit: Revisiting data augmentation and beyond for grammatical error correction. In *Findings of EMNLP*.

Zheng Yuan and Ted Briscoe. 2016. Grammatical error correction using neural machine translation. In *NAACL*.

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 *NAACL*.

Wangchunshu Zhou, Tao Ge, Chang Mu, Ke Xu, Furu Wei, and Ming Zhou. 2020. Improving grammatical error correction with machine translation pairs. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 318–328, Online. Association for Computational Linguistics.

Hyperparam.	Value
Model 1	
Dropout	0.3
Learning rate	5×10^{-5}
Min. learning rate	1×10^{-9}
Init. learning rate	1×10^{-7}
Optimizer	Adam $(0.9, 0.98)$
Max epochs	25
Label smoothing	0.1
Max. tokens	13,000
Seeds	1,2
Model 2	
Dropout	0.1
Learning rate	1×10^{-4}
Optimizer	Adam
Max epochs	5 (20)
Input/output lengths	128
Seeds	42 (1,42)

Table A1: Hyperparameter settings for Model 1 and Model 2. The seeds and the number of epochs are shown separately for the pre-training and the finetuning stages (in parentheses).

GPU	Model type
$A100 \times 4$model 1	3hrs
$A100 \times 4$model 2	12hrs

Table A2: Training times *per epoch* for the pre-training stage (on synthetic data).

A Training Details and Hyperparameters

Hyperparameters Experiments are performed on four A100 32GB GPUs. Hyperparameters for model 1 (we use the Transformer (Vaswani et al., 2017), with the "Transformer (big)" settings and the parameters in Kiyono et al. (2019) for `Pretrain` setting) and for Model 2 are shown in Table A1. Table A2 shows training times per model and per epoch on the synthetic data (15M and 2M sentence pairs for model 1 and model 2, respectively). Finetuning on gold data is fast due to the small sizes of the finetuning sets. We use 2 seeds with each model, and report results averaged over two runs.

B Examples of Russian Learner Errors

Table B3 shows examples of some common Russian learner errors. The top part of the table shows errors on closed-class words (prepositions, conjunctions) and mistakes in inflectional morphology. The bottom part of the table shows lexical mistakes and errors in derivational morphology.

C Synthetic Data Generation Methods

C.1 Baseline Synthetic Data Generation Methods

Random character-level perturbations (Char) This is a simple approach that performs insert/delete/replace operations at the character level. It was used in several GEC works (e.g., Kiyono et al. (2019); Xie et al. (2018)).

Spell-based transformations (Spell) This approach of generating synthetic data showed state-of-the-art performance in English (Bryant et al., 2019; Grundkiewicz and Junczys-Dowmunt, 2019; Grundkiewicz et al., 2019), and other languages (Náplava and Straka, 2019; Flachs et al., 2021). Spell-based confusions include highly confusable words based on edit distance obtained from a dictionary available in a spellchecker. For example, for the target word "there", the confusion set would include words such as *their*, *there're*, *here*. The replacement for the target token is then chosen uniformly at random from its confusion set. Because Aspell is an open-source spellchecker, it is common to use Aspell for this purpose. In line with Grundkiewicz and Junczys-Dowmunt (2019), the word error rate used (percentage of tokens perturbed) is that of 15%, and characters are perturbed in 10% of the word tokens to account for spelling mistakes. We follow Náplava and Straka (2019) for the parameter values for token replacement, deletions, and insertions.

Morphology-based transformations (Morph) In the Morph approach (Choe et al., 2019; Flachs et al., 2021), confusion sets are formed by including all variants that belong to the morphological paradigm of the same base form. An example of a morphological paradigm in English is {"walk", "walking", walked", "walks"} for the base form "walk". In a highly-inflectional language such as Russian, inflectional paradigms exist for nouns, verbs, adjectives, pronouns, and numerals. The paradigms are quite complex, for example, a noun paradigm includes up to 12 wordforms; adjectival paradigms contain 24-26 wordforms, while verb paradigms may include up to 200 forms.

We generate confusion sets based on the output of a morphological analyzer for Russian (Sorokin et al., 2016), applied to large corpus of text that we use to generate synthetic data (see Section 5).[12]

[12] Similar to the spell-based method, 15% of tokens are modified in this approach.

Error type	Example
Mistakes in grammar on closed-class words	
Prep. (ins.,del.,repl.)	в “in” → из “from, out of”
Mistakes in grammar (inflectional morphology)	
Noun:case	специалист-ы “experts” (pl.,nom) → специалист-ам (pl.,dat.)
Verb:number/gender	жив-ут “live” (3rd person pl.) → жив-ет (3rd person sg.)
Verb:aspect	чувствовала “feel” (past, imperf.) → по-чувствовала (past, perf.)
Verb:voice	продолжала “continue” (past, active) → продолжала-сь (past, reflexive)
Mistakes in lexico-morphology category	
Deriv. morph.	вдохнов-ленным “inspired” → вдохнов-енной “inspiring”
Lexical choice	предлагает “proposes” → утверждает “claims”

Table B3: Some common error types in Russian learner data. Partial changes on a word are shown with a hyphen.

For Ukrainian, we follow Flachs et al. (2021) and use Unimorph.

Morph+Spell In this approach, we combine errors generated with the *Morph* method and those generated with the *Spell* method. Flachs et al. (2021) apply this combination approach by combining Aspell and Unimorph, where each method is selected with equal likelihood. Since the two methods do not perform at the same level, we found that it is best not to use equal probabilities. Instead, we use a 15% error rate followed by a 3% error rate for Morph and Spell (for Russian), and for Ukrainian this was reversed. We found these probabilities optimal for the two languages, where the stronger method is assigned a higher value.

D Experiments on Russian

D.1 Additional Results

Key results Tables D6 and D7 show detail Precision, Recall, and $F_{0.5}$ of models trained with and without the SeLex-RT method, for Russian and Ukrainian, respectively.

Effect of SeLex-RT on performance by error type Table D10 shows key results by coarse error type in the Russian RULEC benchmark for the two models.

D.2 Evaluation with Closest Golds

Reference-based evaluation and low-coverage bias The use of too few references in GEC evaluation is known to underestimate system performance and to bias evaluation is known as low-coverage bias (Choshen and Abend, 2018b). The standard approach of evaluating GEC systems is to make use of reference-based measures, where system output is compared against a reference created by a human expert. A system is rewarded for proposing corrections that are in the reference, and penalized for proposing corrections not found in the reference. For this reason, reference-based evaluation measures tend to severely underestimate system performance (Choshen and Abend, 2018b; Mita et al., 2019; Felice and Briscoe, 2015; Bryant and Ng, 2015). When more than a single reference is available, system output is compared independently against each reference, and the best score is selected for each sentence. As a result, scores tend to increase with the number of references used (Ng et al., 2014). Using multiple references thus provides a less biased evaluation of system performance, although does not eliminate the issue entirely.

Bryant and Ng (2015) find that using more than 3 references tends to provide diminishing returns, suggesting that *the use of 3 references may provide a more realistic idea of system performance* (Bryant and Ng, 2015).

Evaluation with Closest Golds Although using multiple references provides a more realistic performance evaluation, the coverage bias is not entirely eliminated. Moreover, performance bias is also error-specific, as errors that allow for a larger set of possible corrections, specifically, lexical mistakes, suffer more than errors such as verb agreement where the number of correction options is limited, as the chance of matching the gold correction for a system is smaller. Indeed, Rozovskaya and Roth (2021) showed that while the performance of GEC systems is severely underestimated with standard evaluations, performance on mistakes that have more correction options (lexical and fluency) is underestimated more severely than performance on spelling and grammar errors. They introduce the concept of evaluations with *closest golds* (CGs), i.e. human references constructed relative to the system-corrected output, akin to post-evaluation for

Target token	Synth. method	Conf. size	Confusion set candidates
продлить “to extend”	Spell	10	продлишь “(you) will extend”; продлит “(he/she) will extend”; пролить “to spill”; подлить “to add (fluid)”
	Morph	48	продлённой “extended (past. partic., sg., fem., instr.)”; продлит “(it/he/she) will extend; продлим “(we) will extend”; продлил “(sg., masc.) extended”; продлены “(they) were extended”; продлен “(it/he/she) was extended”
	SeLex-RT	21	продолжительность “duration”; продлит “(he/she) will extend”; увеличивать “to increase”; повышать “to raise”; продлевают “(they) are extending”; задержать “to delay”
огромного “huge” (sg., masc., gen.)	Spell	6	погромный “pogromous” (sg., masc., nom.); огромный “huge” (sg., masc., nom.); погромного “pogromous” (sg., masc., gen.)
	Morph	38	огромных “huge” (pl., gen.) ; огромным “huge” (sg., masc., instr.); огромной “huge” (sg., fem., gen.); огромнейшей “very huge” (sg., fem., gen.); огромнейшим “very huge” (sg., masc., instr.); огромные “huge” (pl., masc., nom.); огромная “huge” (sg., fem., nom.)
	SeLex-RT	115	огромном “huge” (sg., masc., instr.); богатейшего “rich” ; громадного “enormous”; многочисленных “numerous”; обширными “extensive”; широкого “wide”; крупными “large”; неограниченных “unlimited”; большой “big”; значительных “significant”; большинство “majority”; серьезных “serious”
деньги “money” (pl., gen.)	Spell	5	деньге “money” (sg., fem., dat.); деньгу “money” (sg., fem., acc.) ; деньга ‘money” (sg., fem., nom.);деньки “days” (nom.)
	Morph	10	деньгах “money” (pl., masc., prepos.); деньгам “money” (pl., masc., dat.); деньгами “money” (pl., masc., instr.); денег “money” (pl., masc., gen.)
	SeLex-RT	596	оплату “payment” (acc.); взносы “fees” (nom.); капитала “capital” (gen.); стоимости “cost” (gen.); средств “resources” (gen.); состоянии “fortune” (gen.); сумму “sum” (acc.); заработки “earnings” (nom.); долларов “dollars” (gen.); наличности “cash” (gen.)

Table C4: Sample confusion sets generated with each synthetic method. The tokens in the confusion set are used to corrupt the native data by replacing occurrences of the target word with one of the tokens in the corresponding confusion set.

Original sentence	Как только уходит звание , работа или деньги , уходит и сила . As soon as your title, job or money leaves, your strength leaves, as well .
Char	Как тоьлко уходит звание , работа или деньги , уход ит и сила .
Spell	Как ~~только~~ уходит знание , работа или деньги , уходят и сила .
Morph	Как только уходит звание , работа или деньгами , уходящим и сила .
SeLex-RT	Как только уходит звание , труд или средств , уходит и мощь .

Table C5: An example of an original well-formed sentence (top) from a monolingual Russian corpus and artificially generated erroneous sentences (bottom), using each synthetic data generation method. Modified or replaced tokens are underlined; deleted tokens are crossed out.

Synth.	**RULEC**			**RU-Lang8**		
method	**P**	**R**	**$F_{0.5}$**	**P**	**R**	**$F_{0.5}$**
Model 1 (Baselines)						
Char	50.6	21.7	40.0	50.5	22.2	40.2
Spell	66.0	31.3	54.0	63.0	31.8	52.6
Morph	67.9	35.7	57.5	65.3	35.5	56.0
M+S	**68.8**	**41.2**	**60.7**	**68.9**	**43.1**	**61.6**
SeLex-RT	65.5	24.5	49.1	59.9	26.2	47.7
Model 1 (Best baselines with SeLex-RT)						
M+SeLex-RT	68.7	41.4	60.7	66.7	42.7	59.9
M+S+SeLex-RT	**70.6**	**41.9**	**62.1**	**69.9**	**44.2**	**62.6**
Model 2						
M+S	75.1	36.5	62.0	71.4	36.5	59.9
M+S+SeLex-RT	**75.6**	**37.6**	**62.9**	**71.9**	**37.9**	**60.9**

Table D6: Key results on Russian. Best $F_{0.5}$ and recall for each dataset and model are in bold. *Model 1* denotes models trained from scratch, while *Model 2* refers to models that finetune mT5-Base. *M+S* stands for *Morph+Spell.*

Synth.	**UA-GEC**		
method	**P**	**R**	**$F_{0.5}$**
Model 1 (baselines)			
Char	57.7	47.0	55.2
Spell	66.9	50.3	62.8
Morph	57.5	22.4	43.8
M+S	**67.0**	**51.8**	**63.3**
SeLex-RT	61.9	45.0	57.6
Model 1 (Best baselines+SeLex-RT)			
Spell+SeLex-RT	**68.3**	**53.5**	**64.7**
M+S+SeLex-RT	67.8	53.2	64.3
Model 2			
Spell	73.1	42.9	64.1
Spell+SeLex-RT	**74.6**	**45.4**	**66.1**

Table D7: Key results on Ukrainian. Best $F_{0.5}$ and recall for each model are in bold. *Model 1* denotes models trained from scratch, while *Model 2* refers to models that finetune mT5. *M+S* stands for *Morph+Spell.*

Synth.	**RULEC**			**RU-Lang8**		
method	**P**	**R**	**$F_{0.5}$**	**P**	**R**	**$F_{0.5}$**
3 configurations of the round-trip MT method						
MT only (orig)	37.4	56.0	40.1	40.5	56.5	42.9
Morph+MT	65.4	35.8	56.1	62.4	39.0	55.7
M+S+MT	65.7	39.1	57.8	68.2	42.1	60.7
This work						
M+S+SeLex-RT	**70.6**	**41.9**	**62.1**	**69.9**	**44.2**	**62.6**

Table D8: Comparison with full-sentence round-trip translation method in Lichtarge et al. (2019).

Synthetic	**Performance by error group**														
method	**Grammar**			**Orth.**			**Lex., morph.**			**Other**			**Total**		
	P	**R**	**$F_{0.5}$**	**P**	**R**	**$F_{0.5}$**	**P**	**R**	**$F_{0.5}$**	**P**	**R**	**$F_{0.5}$**	**P**	**R**	**$F_{0.5}$**
Model 1 (baselines)															
Spell	74.8	28.5	56.4	69.3	50.4	64.5	44.0	3.7	13.8	26.7	15.2	23.2	63.0	31.8	52.6
Morph	70.2	39.0	60.5	71.6	53.9	67.2	27.5	1.7	6.9	27.3	10.1	20.3	65.3	35.5	56.0
M+S	78.8	48.0	69.8	75.9	62.0	72.6	38.9	4.5	15.4	27.9	15.1	23.9	69.5	41.4	61.2
Model 1 (baselines+SeLex-RT)															
SeLex-RT	69.0	25.2	51.2	67.2	41.3	59.7	26.6	10.5	20.3	25.2	7.5	17.1	59.9	26.2	47.7
M+S+SeLex-RT	80.6	48.2	71.1	76.5	62.4	73.2	41.3	14.2	29.9	30.3	14.5	24.9	70.3	44.3	62.9
Model 2 (best baseline+SeLex-RT)															
M+S	79.6	41.5	67.2	73.9	52.1	68.2	58.3	6.6	22.8	33.3	10.9	23.7	71.3	36.5	59.9
M+S+SeLex-RT	81.4	43.6	69.4	74.7	53.0	69.0	47.7	9.9	27.0	32.1	10.0	22.3	71.8	37.9	60.9

Table D9: Key results by coarse error group on Russian (RU-Lang8).

Synthetic	**Performance by error group**														
method	**Grammar**			**Orth.**			**Lex., morph.**			**Other**			**Total**		
	P	**R**	**$F_{0.5}$**	**P**	**R**	**$F_{0.5}$**	**P**	**R**	**$F_{0.5}$**	**P**	**R**	**$F_{0.5}$**	**P**	**R**	**$F_{0.5}$**
Model 1 (baselines)															
Spell	79.1	31.3	60.6	70.4	49.9	65.1	45.2	2.4	9.9	20.8	9.1	16.6	66.0	31.3	54.0
Morph	69.2	39.7	60.3	74.3	55.4	69.6	29.8	2.3	8.9	18.7	4.3	11.1	67.9	35.7	57.5
M+S	76.3	47.4	68.0	73.9	60.8	70.9	40.4	3.4	12.6	15.9	6.1	12.1	68.8	41.2	60.7
Model 1 (baselines+SeLex-RT)															
SeLex-RT	74.5	23.7	52.1	72.8	41.0	63.0	31.8	8.3	20.3	14.2	2.8	7.9	65.5	24.5	49.1
M+S+LexRT	78.7	47.9	69.8	76.6	60.6	72.8	36.2	10.0	23.8	19.5	6.3	13.7	71.0	42.0	62.3
Model 2 (best baseline+SeLex-RT															
M+S	81.3	45.4	70.2	78.1	50.2	70.3	49.3	5.9	19.8	28.6	6.5	16.9	75.1	36.5	62.0
M+S+LexRT	81.6	45.6	70.5	79.4	52.0	71.9	50.0	8.1	24.7	29.3	6.9	17.7	75.6	37.6	62.9

Table D10: Key results by coarse error group on Russian (RULEC).

machine translation.

Similarly, for GEC, Rozovskaya and Roth (2021) generate references relative to the system outputs and not the original texts and show that automatic evaluation against reference golds (RGs), that is corrections generated relative to the original sentences, severely underestimates model performance. This is because the set of possible corrections for a given source sentence is extremely large and possibly infinite. We expect that evaluating against closest golds, i.e. corrections produced relative to the system hypothesis, would give us the most realistic evaluation of the system quality. To provide a more accurate evaluation of the SeLex-RT component, we apply this method to evaluate the contribution of the SeLex-RT approach to handle lexical mistakes. We generate CGs for the same subset of 500 sentences for 4 system outputs – Morph, Spell, and Morph+Spell, and Morph+Spell+SeLex-RT, following the same approach outlined in Rozovskaya and Roth (2021). The annotations are generated by one of the raters who contributed to the original annotation of RULEC. The annotators were compensated $25 per hour for the work. Complete results are shown in Tables D12 and D13 for RULEC and RU-Lang8, respectively. The CG annotations will be released upon paper publication.

Example comparing evaluation using reference golds and closest golds In Table D11 we show an example (in English) that shows two references (ref. 1 and ref. 2). Ref. 1 is the original reference (created relative to the source sentence) and ref. 2 is generated relative to system hypothesis.

Note that system scores (P,R, $F_{0.5}$) depend on the amount of *overlap* between a reference and system hypothesis. Evaluation wit closest golds attempts to generate a reference that is as close as possible to the system hypothesis (by producing a reference relative to system output, and not relative to the original sentence). Thus, evaluation with CGs provides a more realistic evaluation of system performance. Note that the amount of overlap (number of correct edits) is larger for ref. 2, and thus the F0.5 score against ref. 2 is higher (91.0) vs. F0.5 score against ref. 1 (50.0):

D.3 Comparison to Related Work

We compare our models with prior work using the original single-reference annotations for RULEC and RU-Lang8 in Table D14. The top segment of the table shows models trained in this work. The second segment shows models that are directly comparable to ours due to similar architectures and/or model parameters. We outperform all models in segment 2 (with the exception of a much more powerful mT5-Large pre-trained on 10M sentence pairs), often by a large margin, even when we use less synthetic data. Interestingly, a recent implementation of tagged corruption models (Stahlberg and Kumar, 2024) (S&K'24) present a similar model to ours but only obtains a result of 26.4 compared to 51.2 on RULEC. This suggests the importance of high-quality synthetic data even when using pre-trained models as a starting point in GEC.

The remaining three segments show results of previous work broken down by the amount of gold data used in training and fine-tuning. The special symbols next to each model indicate the type and amount of gold data used (explained in the table caption). Our mT5-base result is comparable to gT5 xxl (13B parameters, last table section); with mT5-large, we obtain a 2-point improvement. Our smaller seq2seq model outperforms all models of similar sizes (section 2 in the table) that also use RULEC training data. Sorokin (2022) uses ruGPT-3 and RoBERTa-large. Their model is comparable to mT5-large, in terms of parameters, but is trained on Russian data, whereas mT5 is multilingual.

E Analysis of the SeLex-RT Method

Table D15 shows examples of errors missed by the baseline models but corrected by models that use the BT component.

Source	The settings are very reallistic and the actors had a great performance .
System hypo System edits	The settings are very realistic and the actors had great performance . reallistic → realistic; had a great → had great
Evaluation against original gold (OG)	
Ref. 1 (OG)	The settings are very realistic and the actors gave a great performance .
Gold edits (OG)	(1) reallistic → realistic; (2) had → gave
Correct edits (OG)	(1) reallistic → realistic
Performance against OG	P = 50.0; R = 50.0; $F_{0.5}$ = 50.0
Evaluation against closest gold (CG)	
Ref. 2 (CG)	The settings are very realistic and the actors had great performances .
Gold edits (CG)	(1) reallistic → realistic; (2) had great → had a great; (3) performance → performances
Performance against CG	P = 100.0; R = 66.0; $F_{0.5}$ = **91.0**

Table D11: Evaluation with original reference (OG) and closest gold reference (CG).

Synth. data	**Ref. set**	**Performance**								
		Grammar			**Lex.,morph.**			**Total**		
		P	**R**	**$F_{0.5}$**	**P**	**R**	**$F_{0.5}$**	**P**	**R**	**$F_{0.5}$**
Model 1 (Baselines)										
Spell	3 RGs	82.0	33.1	63.3	50.0	3.4	13.3	75.9	33.9	60.8
	CG	87.1	40.0	70.5	66.7	4.0	16.1	85.7	43.3	71.6
Morph	3 RGs	68.6	38.0	59.1	36.4	3.4	12.5	71.0	38.6	60.8
	CG	83.8	51.8	74.6	90.9	8.4	30.7	85.6	50.8	75.3
M+S	3 RGs	75.8	45.1	66.7	53.3	6.7	22.3	74.0	43.9	65.0
	CG	85.9	57.4	78.1	75.0	10.5	33.7	86.5	56.9	78.3
Model 1 (M+S+SeLex-RT)										
M+S+SeLex-RT	3 RGs	74.9	46.6	66.8	44.4	15.3	32.2	72.3	44.7	64.3
	CG	86.6	60.2	79.6	80.4	**29.3**	**59.6**	87.6	**58.7**	**79.7**

Table D12: Evaluation on RULEC with three references (RGs) and closest golds (CGs) on a 500-sentence subset for baselines (Model 1) and the model that includes the SeLex-RT component. Best overall recall and $F_{0.5}$, as well as best values on the lexico-morph. errors are in bold.

Synth. data	**Ref. set**	**Performance**								
		Grammar			**Lex.,morph.**			**Total**		
		P	**R**	**$F_{0.5}$**	**P**	**R**	**$F_{0.5}$**	**P**	**R**	**$F_{0.5}$**
Model 1 (Baselines)										
Spell	3 RGs	87.9	26.3	59.9	57.1	4.3	16.4	66.0	28.5	52.3
	CG	91.1	30.5	65.2	66.7	5.6	20.8	76.5	35.2	61.9
Morph	3 RGs	73.2	35.2	60.2	77.8	4.0	16.7	67.8	32.8	55.9
	CG	82.7	43.8	70.2	100.0	5.5	22.5	78.9	41.5	66.8
M+S	3 RGs	84.9	48.7	73.9	64.3	5.0	19.0	73.7	40.5	63.3
	CG	91.0	57.4	81.5	84.6	**6.7**	**25.5**	82.4	49.3	**72.6**
Model 1 (M+S+SeLex-RT)										
M+S+SeLex-RT	3 RGs	84.7	47.9	73.4	42.4	15.3	31.3	72.0	42.1	63.0
	CG	91.7	56.5	81.5	74.6	**28.3**	**56.3**	83.4	**52.4**	**74.6**

Table D13: Evaluation on RU-Lang8 with three references (RGs) and closest golds (CGs) on a 500-sentence subset for baselines (Model 1) and the model that includes the SeLex-RT component. Best overall recall and $F_{0.5}$, as well as best values on the lexico-morph. errors are in bold.

Model	**F$_{0.5}$**	
	RULEC	RU-Lang8
This work (M+S+SeLex-RT)		
Model 1 (15M synth.) ★	49.2	51.8
Model 2 (2M synth.)★	51.2	52.3
Previous work (similar model arch. to ours)		
P&R (15M synth., cf. 1) ★	47.4	47.7
P&R (10M, mT5-Base) ★	51.0	49.8
P&R (10M, mT5-Large) ★	53.2	54.5
S&K mT5-Base, 2.5M ★	26.4	-
S&K mT5-XXL 2.5M ★	44.3	-
Rothe et al. (2021) gT5 base ★	26.2	-
Náplava and Straka (2019) ★	47.2	-
Flachs et al. (2021) ★	44.7	-
Katsumata and Komachi (2020) ★	44.4	-
Náplava and Straka (2019) ✷	50.2	-
Rothe et al. (2021) gT5 xxl ✦	51.6	-
Sorokin (2022) ‘scorer-only’ ✦	53.4	-
Sorokin (2022) ‘combined’ ✦	55.0	-

Table D14: Comparison with previous work for Russian, using original references. The top segment shows models trained in this work. The second segment shows previous work with similar model architectures to ours. The remaining segments show results obtained in previous work, broken down by the amount of gold data used. Extra large models are grouped in the bottom segment. ★ refers to models that use RULEC training data for fine-tuning. ✷ denotes models that use RULEC training and dev data for fine-tuning; ✦ denotes extra large models in terms of parameters and native data used that also use RULEC training data. *S&K* stands for Stahlberg and Kumar (2024). P&R stands for Palma Gomez and Rozovskaya (2024).

Технологическая граница значит?/означает границу сегодняшнего развития технологии. “Technological boundary **means?/signifies** the limit of today technological development.”
Мне также надо быть внимательной и почтительной и следовать?/соблюдать важные обычаи . “I also need to pay attention, be respectful, and **follow?/observe** important traditions.”
Я провела два года в качестве учителя английского языка в Казахстане , ...но я чувствую факт , что у меня мало обучения?/знаний быть преподавателем . “I spent two years as a teacher of English in Kazakhstan but I feel that I do not have enough **learning?/knowledge** to be a teacher.”
Но сегодня я поинтересовалась разницей норм?/правил эвакуации между двумя авариями в Фукусиме и Чернобыле . “But today I asked about the difference in the evacuation **norms?/rules** for two accidents – Fukusima and Chenobyl.”
Несколько?/Некоторые из их стали диверсифицировать в другие продукты... “**Some?/Several** of them started changing into a different type of products...”
В такой ситуации можно верить только простым честным людям , которые сейчас есть?/находятся рядом , а ни в коем случае не правительству... “In such a situation, one can only trust simple honest people who **exist?/are** there for you, and neve the government...”

Table D15: Examples of Russian lexical errors missed by standard models but corrected by models trained on data that includes the SeLex-RT-based confusions.