Grammatical Error Correction in Low-Resource Scenarios

Jakub Náplava and **Milan Straka**
Charles University,
Faculty of Mathematics and Physics,
Institute of Formal and Applied Linguistics
{naplava,straka}@ufal.mff.cuni.cz

Abstract

Grammatical error correction in English is a long studied problem with many existing systems and datasets. However, there has been only a limited research on error correction of other languages. In this paper, we present a new dataset AKCES-GEC on grammatical error correction for Czech. We then make experiments on Czech, German and Russian and show that when utilizing synthetic parallel corpus, Transformer neural machine translation model can reach new state-of-the-art results on these datasets. AKCES-GEC is published under CC BY-NC-SA 4.0 license at http://hdl.handle.net/11234/1-3057, and the source code of the GEC model is available at https://github.com/ufal/low-resource-gec-wnut2019.

1 Introduction

A great progress has been recently achieved in grammatical error correction (GEC) in English. The performance of systems has since CoNLL 2014 shared task (Ng et al., 2014) increased by more than 60% on its test set (Bryant et al., 2019) and also a variety of new datasets appeared. Both rule-based models, single error-type classifiers and their combinations were due to larger amount of data surpassed by statistical and later by neural machine translation systems. These address GEC as a translation problem from a language of ungrammatical sentences to a grammatically correct ones.

Machine translation systems require large amount of data for training. To cope with this issue, different approaches were explored, from acquiring additional corpora (e.g. from Wikipedia edits) to building a synthetic corpus from clean monolingual data. This was apparent on recent Building Educational Applications (BEA) 2019 Shared Task on GEC (Bryant et al., 2019) when top scoring teams extensively utilized synthetic corpora.

The majority of research has been done in English. Unfortunately, there is a limited progress on other languages. Namely, Boyd (2018) created a dataset and presented a GEC system for German, Rozovskaya and Roth (2019) for Russian, Náplava (2017) for Czech and efforts to create annotated learner corpora were also done for Chinese (Yu et al., 2014), Japanese (Mizumoto et al., 2011) and Arabic (Zaghouani et al., 2015).

Our contributions are as follows:

- We introduce a new Czech dataset for GEC. In comparison to dataset of Šebesta et al. (2017) it contains separated edits together with their type annotations in M2 format (Dahlmeier and Ng, 2012) and also has two times more sentences.
- We extend the GEC model of Náplava and Straka (2019) by utilizing synthetic training data, and evaluate it on Czech, German and Russian, achieving state-of-the-art results.

2 Related Work

There are several main approaches to GEC in *low-resource* scenarios. The first one is based on a noisy channel model and consists of three components: a candidate model to propose (word) alternatives, an error model to score their likelihood and a language model to score both candidate (word) probability and probability of a whole new sentence. Richter et al. (2012) consider for a given word all its small modifications (up to character edit distance 2) present in a morphological dictionary. The error model weights every character edit by a trained weight, and three language models (for word forms, lemmas and POS tags) are used to choose the most probable sequence of corrections. A candidate model of Bryant and Briscoe

Proceedings of the 2019 EMNLP Workshop W-NUT: The 5th Workshop on Noisy User-generated Text, pages 346–356
Hong Kong, Nov 4, 2019. ©2019 Association for Computational Linguistics

(2018) contains for each word spell-checker proposals, its morphological variants (if found in Automatically Generated Inflection Database) and, if the word is either preposition or article, also a set of predefined alternatives. They assign uniform probability to all changes, but use strong language model to re-rank all candidate sentences. Lacroix et al. (2019) also consider single word edits extracted from Wikipedia revisions.

Other popular approach is to extract parallel sentences from Wikipedia revision histories. A great advantage of such an approach is that the resulting corpus is, especially for English, of great size. However, as Wikipedia edits are not human curated specifically for GEC edits, the corpus is extremely noisy. Grundkiewicz and Junczys-Dowmunt (2014) filter this corpus by a set of regular expressions derived from NUCLE training data and report a performance boost in statistical machine translation approach. Grundkiewicz et al. (2019) filter Wikipedia edits by a simple language model trained on BEA 2019 development corpus. Lichtarge et al. (2019), on the other hand, reports that even without any sophisticated filtering, Transformer (Vaswani et al., 2017) can reach surprisingly good results when used iteratively.

The third approach is to create synthetic corpus from a clean monolingual corpus and use it as additional data for training. Noise is typically introduced either by rule-based substitutions or by using a subset of the following operations: token replacement, token deletion, token insertion, multi-token swap and spelling noise introduction. Yuan and Felice (2013) extract edits from NUCLE and apply them on a clean text. Choe et al. (2019) apply edits from W&I+Locness training set and also define manual noising scenarios for preposition, nouns and verbs. Zhao et al. (2019) use an unsupervised approach to synthesize noisy sentences and allow deleting a word, inserting a random word, replacing a word with random word and also shuffling (rather locally). Grundkiewicz et al. (2019) improve this approach and replace a token with one of its spell-checker suggestions. They also introduce additional spelling noise.

3 Data

In this Section, we present existing corpora for GEC, together with newly released corpus for Czech.

3.1 AKCES-GEC

The AKCES (Czech Language Acquisition Corpora; Šebesta, 2010) is an umbrella project comprising of several acquisition resources – CzeSL (learner corpus of Czech as a second language), ROMi (Romani ethnolect of Czech Romani children and teenagers) and SKRIPT and SCHOLA (written and spoken language collected from native Czech pupils, respectively).

We present the AKCES-GEC dataset, which is a grammar error correction corpus for Czech generated from a subset of AKCES resources. Concretely, the AKCES-GEC dataset is based on CzeSL-man corpus (Rosen, 2016) consisting of manually annotated transcripts of essays of non-native speakers of Czech. Apart from the released CzeSL-man, AKCES-GEC further utilizes additional unreleased parts of CzeSL-man and also essays of Romani pupils with Romani ethnolect of Czech as their first language.

The CzeSL-man annotation consists of three Tiers – Tier 0 are transcribed inputs, followed by the level of orthographic and morphemic corrections, where only word forms incorrect in any context are considered (Tier 1). Finally, the rest of errors is annotated at Tier 2. Forms at different Tiers are manually aligned and can be assigned one or more error types (Jelínek et al., 2012). An example of the annotation is presented in Figure 1, and the list of error types used in CzeSL-man annotation is listed in Table 1.

We generated AKCES-GEC dataset using the three Tier annotation of the underlying corpus. We employed Tier 0 as source texts, Tier 2 as corrected texts, and created error edits according to the manual alignments, keeping error annotations where available.[1] Considering that the M2 format (Dahlmeier and Ng, 2012) we wanted to use does not support non-local error edits and therefore cannot efficiently encode word transposition on long distances, we decided to consider word swaps over at most 2 correct words a single edit (with the constant 2 chosen according to the coverage of long-range transpositions in the data). For illustration, see Figure 2.

The AKCES-GEC dataset consists of an explicit train/development/test split, with each set divided into foreigner and Romani students; for de-

[1] The error annotations are unfortunately not available in the whole underlying corpus, and not all errors are annotated with at least one label.

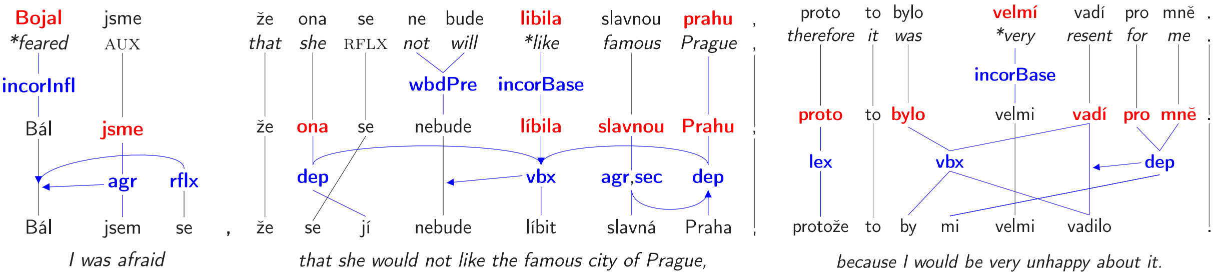
Bojal jsme , že ona se ne bude líbila slavnou prahu , proto to bylo velmí vadí pro mně .
*feared AUX that she RFLX not will *like famous Prague therefore it was *very resent for me
incorInfl wbdPre incorBase incorBase
Bál jsme že ona se nebude líbila slavnou Prahu , proto to bylo velmi vadí pro mně .
agr rflx dep vbx agr,sec dep lex vbx dep
Bál jsem se , že se jí nebude líbit slavná Praha , protože to by mi velmi vadilo .
I was afraid
that she would not like the famous city of Prague,
because I would be very unhappy about it.

Figure 1: Example of two-level annotation of a sentence in CzeSL corpus, reproduced from (Rosen, 2016).

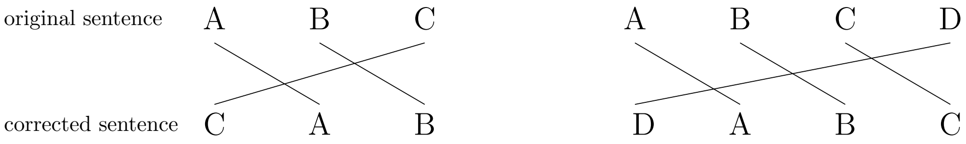

Figure 2: Word swap over one or two correct words (on the left) is considered a single edit (A B C→ C A B). Word swap over more than two correct words (on the right) is represented as two edits of deleting D and inserting D.

Error type	Description	Example	Occ
incorInfl	incorrect inflection	[**pracovají** → *pracují*] v továrně	8 986
incorBase	incorrect word base	musíš to [**posvětlit** → *posvětit*]	20 334
fwFab	non-emendable, „fabricated" word	pokud nechceš slyšet [**smášky**]	78
fwNC	foreign word	váza je na [**Tisch** → *stole*]	166
flex	supplementary flag used with fwFab and fwNC marking the presence of inflection	jdu do [**shopa** → *obchodu*]	34
wbdPre	prefix separated by a space or preposition w/o space	musím to [**při pravit** → *připravit*]	817
wbdComp	wrongly separated compound	[**český anglický** → *česko-anglický*] slovník	92
wbdOther	other word boundary error	[**mocdobře** → *moc dobře*]; [**atak** → *a tak*]	1326
stylColl	colloquial form	[**dobrej** → *dobrý*] film	3 533
stylOther	bookish, dialectal, slang, hyper-correct form	holka s [**hnědými očimi** → *hnědýma očima*]	156
agr	violated agreement rules	to jsou [**hezké** → *hezcí*] chlapci; Jana [**čtu** → *čte*]	5 162
dep	error in valency	bojí se [**pes** → *psa*]; otázka [**čas** → *času*]	6 733
ref	error in pronominal reference	dal jsem to jemu i [**jejího** → *jeho*] bratrovi	344
vbx	error in analytical verb form or compound predicate	musíš [**přijdeš** → *přijít*]; kluci [**jsou**] běhali	864
rflx	error in reflexive expression	dívá [∅ → *se*] na televizi; Pavel [**si** → *se*] raduje	915
neg	error in negation	[**půjdu ne** → *nepůjdu*] do školy	111
lex	error in lexicon or phraseology	dopadlo to [**přírodně** → *přirozeně*]	3 967
use	error in the use of a grammar category	pošta je [**nejvíc blízko** → *nejblíže*]	1 458
sec	secondary error (supplementary flag)	stará se o [**našich holčičkách** → *naše holčičky*]	866
stylColl	colloquial expression	viděli jsme [**hezký** → *hezké*] holky	3 533
stylOther	bookish, dialectal, slang, hyper-correct expression	rozbil se mi [**hadr**]	156
stylMark	redundant discourse marker	[**no**]; [**teda**]; [**jo**]	15
disr	disrupted construction	známe [**hodné spoustu** → *spoustu hodných*] lidí	64
problem	supplementary label for problematic cases		175
unspec	unspecified error type		69 123

Table 1: Error types used in CzeSL corpus taken from (Jelínek et al., 2012), including number of occurrences in the dataset being released. Tier 1 errors are in the upper part of the table, Tier 2 errors are in the lower part. The *stylColl* and *stylOther* are annotated on both Tiers, but we do not distinguish on which one in the AKCES-GEC.

velopment and test sets, the foreigners are further split into Slavic and non-Slavic speakers. Furthermore, the development and test sets were annotated by two annotators, so we provide two references if the annotators utilized the same sentence segmentation and produced different annotations.

The detailed statistics of the dataset are presented in Table 2. The AKCES-GEC dataset is released under the CC BY-NC-SA 4.0 license at http://hdl.handle.net/11234/1-3057.

We note that there already exists a CzeSL-GEC dataset (Šebesta et al., 2017). However, it consists only of a subset of data and does not contain error types nor M2 files with individual edits.

3.2 English

Probably the largest corpus for English GEC is the Lang-8 Corpus of Learner English (Mizumoto

		Train				Dev				Test			
		Doc	Sent	Word	Error r.	Doc	Sent	Word	Error r.	Doc	Sent	Word	Error r.
Foreign.	Slavic	1 816	27 242	289 439	22.2 %	70	1 161	14 243	21.8 %	69	1 255	14 984	18.8 %
	Other					45	804	8 331	23.8 %	45	879	9 624	20.5 %
Romani		1 937	14 968	157 342	20.4 %	80	520	5 481	21.0 %	74	542	5 831	17.8 %
Total		3 753	42 210	446 781	21.5 %	195	2 485	28 055	22.2 %	188	2 676	30 439	19.1 %

Table 2: Statistics of the AKCES-GEC dataset – number of documents, sentences, words and error rates.

et al., 2011; Tajiri et al., 2012). It comes from an online language learning website, where users are able to post texts in language they are learning. These texts then appear to native speakers for correction. The corpus has over 100 000 raw English entries comprising of more than 1M sentences. Due to the fact that texts are corrected by online users, this corpus is also quite noisy.

Other corpora are corrected by trained annotators making them much cleaner but also significantly smaller. NUCLE (Dahlmeier et al., 2013) has 57 151 sentences originating from 1 400 essays written by mainly Asian undergraduate students at the National University of Singapore. FCE (Yannakoudakis et al., 2011) is a subset of the Cambridge Learner Corpus (CLC) and has 33 236 sentences from 1 244 written answers to FCE exam questions. Recent Write & Improve (W&I) and LOCNESS v2.1 (Bryant et al., 2019; Granger, 1998) datasets were annotated for different English proficiency levels and a part of them also comes from texts written by native English speakers. Altogether, it has 43 169 sentences.

To evaluate system performance, CoNLL-2014 test set is most commonly used. It comprises of 1 312 sentences written by 25 South-East Asian undergraduates. The gold annotations are matched against system hypothesis using MaxMatch scorer outputting $F_{0.5}$ score. The other frequently used dataset is JFLEG (Napoles et al., 2017; Heilman et al., 2014), which also tests systems for how fluent they sound by utilizing the GLEU metric (Napoles et al., 2015). Finally, recent W&I and LOCNESS v2.1 test set allows to evaluate systems on different levels of proficiency and also against different error types (utilizing ERRANT scorer).

3.3 German

Boyd (2018) created GEC corpus for German from two German learner corpora: Falko and MERLIN (Boyd et al., 2014). The resulting dataset comprises of 24 077 sentences divided into training, development and test set in the ratio of 80:10:10. To evaluate system performance, MaxMatch scorer is used.

Apart from creating the dataset, Boyd (2018) also extended ERRANT for German. She defined 21 error types (15 based on POS tags) and extended spaCy[2] pipeline to classify them.

3.4 Russian

Rozovskaya and Roth (2019) introduced RULEC-GEC dataset for Russian GEC. To create this dataset, a subset of RULEC corpus with foreign and heritage speakers was corrected. The final dataset has 12 480 sentences annotated with 23 error tags. The training, development and test sets contain 4 980, 2 500 and 5 000 sentence pairs, respectively.

3.5 Corpora Statistics

Table 3 indicates that there is a variety of English datasets for GEC. As Náplava and Straka (2019) show, training Transformer solely using these annotated data gives solid results. On the other hand, there is only limited number of data for Czech, German and Russian and also the existing systems perform substantially worse. This motivates our research in these low-resource languages.

Table 3 also presents an average error rate of each corpus. It is computed using maximum alignment of original and annotated sentences as a ratio of non-matching alignment edges (insertion, deletion, and replacement). The highest error rate of 21.4 % is on Czech dataset. This implies that circa every fifth word contains an error. German is also quite noisy with an error rate of 16.8 %. The average error rate on English ranges from 6.6 % to 14.1 % and, finally, the Russian corpus contains the least errors with an average error rate of 6.4%.

[2] https://spacy.io/

Language	Corpus	Sentences	Err. r.
English	Lang-8	1 147 451	14.1%
	NUCLE	57 151	6.6%
	FCE	33 236	11.5%
	W&I+LOCNESS	43 169	11.8%
Czech	AKCES-GEC	42 210	21.4%
German	Falko-MERLIN	24 077	16.8%
Russian	RULEC-GEC	12 480	6.4%

Table 3: Statistics of available corpora for Grammatical Error Correction.

3.6 Tokenization

The most popular metric for benchmarking systems are MaxMatch scorer (Dahlmeier and Ng, 2012) and ERRANT scorer (Bryant et al., 2017). They both require data to be tokenized; therefore, most of the GEC datasets are tokenized.

To tokenize monolingual English and German data, we use spaCy v1.9.0 tokenizer utilizing *en_core_web_sm-1.2.0* and *de* model. We use custom tokenizers for Czech[3] and Russian[4].

4 System Overview

We use neural machine translation approach to GEC. Specifically, we utilize Transformer model (Vaswani et al., 2017) to translate ungrammatical sentences to grammatically correct ones. We further follow Náplava and Straka (2019) and employ source and target word dropouts, edit-weighted MLE and checkpoint averaging. We do not use iterative decoding in this work, because it substantially slows down decoding. Our models are implemented in Tensor2Tensor framework version 1.12.0.[5]

4.1 Pretraining on Synthetic Dataset

Due to the limited number of annotated data in Czech, German and Russian we decided to create a corpus of synthetic parallel sentences. We were also motivated by the fact that such approach was shown to improve performance even in English with substantially more annotated training data.

We follow Grundkiewicz et al. (2019), who use an unsupervised approach to create noisy input sentences. Given a clean sentence, they sample a probability p_{err_word} from a normal distribution with a predefined mean and a standard deviation. After multiplying p_{err_word} by a number of words in the sentence, as many sentence words are selected for modification. For each chosen word, one of the following operations is performed with a predefined probability: substituting the word with one of its ASpell[6] proposals, deleting it, swapping it with its right-adjacent neighbour or inserting a random word from dictionary after the current word. To make the system more robust to spelling errors, same operations are also used on individual characters with p_{err_char} sampled from a normal distribution with a different mean and standard deviation than p_{err_word} and (potentially) different probabilities of character operations.

When we inspected the results of a model trained on such dataset in Czech, we observed that the model often fails to correct casing errors and sometimes also errors in diacritics. Therefore, we extend word-level operations to also contain operation to change casing of a word. If a word is chosen for modification, it is with 50% probability whole converted to lower-case, or several individual characters are chosen and their casing is inverted. To increase the number of errors in diacritics, we add a new character-level noising operation, which for a selected character either generates one of its possible diacritized variants or removes diacritics. Note that this operation is performed only in Czech.

We generate synthetic corpus for each language from WMT News Crawl monolingual training data (Bojar et al., 2017). We set p_{err_word} to 0.15, p_{err_char} to 0.02 and estimate error distributions of individual operations from development sets of each language. The constants used are presented in Table 4. We limited amount of synthetic sentences to 10M in each language.

4.2 Finetuning

A model is (pre-)trained on a synthetic dataset until convergence. Afterwards, we finetune the model on a mix of original language training data and synthetic data. When finetuning the model, we preserve all hyperparameters (e.g., learning rate and optimizer moments). In other words, the training continues and only the data are replaced.

When finetuning, we found that it is crucial to preserve some portion of synthetic data in the training corpus. Finetuning with original training

[3] A slight modification of MorphoDiTa tokenizer.

[4] https://github.com/aatimofeev/spacy_russian_tokenizer

[5] https://github.com/tensorflow/tensor2tensor

[6] http://aspell.net/

Language	Token-level operations					Character-level operations				
	sub	ins	del	swap	recase	sub	ins	del	recase	toggle diacritics
English	0.6	0.2	0.1	0.05	0.05	0.25	0.25	0.25	0.25	0
Czech	0.7	0.1	0.05	0.1	0.05	0.2	0.2	0.2	0.2	0.2
German	0.64	0.2	0.1	0.01	0.05	0.25	0.25	0.25	0.25	0
Russian	0.65	0.1	0.1	0.1	0.05	0.25	0.25	0.25	0.25	0

Table 4: Language specific constants for token- and character-level noising operations.

data leads to fast overfitting with worse results on all of Czech, German and Russian. We also found out that it also slightly helps on English.

We ran a small grid-search to estimate the ratio of synthetic versus original sentences in the finetuning phase. Although the ratio of 1:2 (5M original oversampled training pairs and 10M synthetic pairs) still overfits, we found it to work best for English, Czech and German, and stop training when the performance on the development set starts deteriorating. For Russian, the ratio of 1:20 (0.5M oversampled training pairs and 10M synthetic pairs) works the best.

The original sentences for English finetuning are concatenated sentences from Lang-8 Corpus of Learner English, FCE, NUCLE and W&I and LOCNESS. To better match domain of test data, we oversampled training set by adding W&I training data 10 times, FCE data 5 times and NUCLE corpus 5 times to the training set. The original sentences in Czech, German and Russian are the training data of the corresponding languages.

4.3 Implementation Details

When running grid search for hyperparameter tuning, we use *transformer_base_single_gpu* configuration, which uses only 1 GPU to train *Transformer Base* model. After we select all hyperparameter, we train *Transformer Big* architecture on 4 GPUs. Hyperparameters described in following paragraphs belong to both architectures.

We use Adafactor optimizer (Shazeer and Stern, 2018), linearly increasing the learning rate from 0 to 0.011 over the first 8000 steps, then decrease it proportionally to the number of steps after that (using the `rsqrt_decay` schedule). Note that this only applies to the pre-training phase.

All systems are trained on Nvidia P5000 GPUs. The vocabulary consists of approximately 32k most common word-pieces, the batch size is 2000 word-pieces per each GPU and all sentences with more than 150 word-pieces are discarded during training. Model checkpoints are saved every hour.

At evaluation time, we decode using a beam size of 4. Beam-search length-balance decoding hyperparameter alpha is set to 0.6.

5 Results

We present results of our model when trained on English, Czech, German and Russian in this Section. As we are aware of only one system in German, Czech and Russian to compare with, we start with English model discussion. We show that our model is on par or even slightly better than current state-of-the-art systems in English when no ensembles are allowed. We then discuss our results on other languages, where our system exceeds all existing systems by a large margin.

In all experiments, we report results of three systems: *synthetic pretrain*, which is based on Transformer Big and is trained using synthetic data only, and *finetuned* and *finetuned base single GPU*, which are based on Transformer Big and Base, respectively, and are both pretrained and finetuned. Note that even if the *finetuned base* system has 3 times less parameters than *finetuned*, its results on some languages are nearly identical.

We also tried training the system using annotated data only. With our model architecture, all but English experiments (which contain substantially more data) starts overfitting quickly, yielding poor performance. The overfitting problem could be possibly addressed as proposed by Sennrich and Zhang (2019). Nevertheless, given that our best system on English is by circa 10 points in $F_{0.5}$ score better than the system trained solely on annotated data, we focused primarily on the synthetic data experiments.

Apart from the W&I+L development and test sets, which are evaluated using ERRANT scorer, we use MaxMatch scorer in all experiments.

System	W&I+L test	W&I+L dev	CoNLL 14 test	
			No W&I+L	With W&I+L
including ensembles				
Lichtarge et al. (2019)	–	–	60.40	–
Zhao et al. (2019)	–	–	61.15	–
Xu et al. (2019)	67.21	55.37	–	63.20
Choe et al. (2019)	69.06	52.79	57.50	–
Grundkiewicz et al. (2019)	**69.47**	53.00	**61.30**	**64.16**
no ensembles				
Lichtarge et al. (2019)	–	–	**56.80**	–
Xu et al. (2019)	**63.94**	52.29	–	60.90
Choe et al. (2019)	63.05	47.75	–	–
Grundkiewicz et al. (2019)	–	50.01	–	–
no ensembles				
Our work – synthetic pretrain	51.16	32.76	41.85	44.12
Our work – finetuned base single GPU	67.18	52.80	59.87	–
Our work – finetuned	69.00	53.30	60.76	63.40

Table 5: Comparison of systems on two English GEC datasets. CoNLL 2014 Test Set is divided into two system groups (columns): those who do not train on W&I+L training data and those who do.

System	P	R	$F_{0.5}$
Boyd (2018)	51.99	29.73	45.22
Our work – synthetic pretrain	67.45	26.35	51.41
Our work – finetuned base single GPU	78.11	59.13	73.40
Our work – finetuned	78.21	59.94	73.71

Table 6: Results on on Falko-Merlin Test Set (German).

5.1 English

We provide comparison between our model and existing systems on W&I+L test and development sets and on CoNLL 14 test set in Table 5. Even if the results on the W&I+L development set are only partially indicative of system performance, we report them due to the W&I+L test set being blind. All mentioned papers do not train their systems on the development set, but use it only for model selection. Also note that we split the results on CoNLL 14 test set into two groups: those who do not use the W&I+L data for training, and those who do. This is to allow a fair comparison, given that the W&I+L data were not available before the BEA 2019 Shared Task on GEC.

The best performing systems are utilizing ensembles. Table 5 shows an evident performance boost (3.27-6.01 points) when combining multiple models into an ensemble. The best performing system on English is an ensemble system of Grundkiewicz et al. (2019).

The aim of this paper is to concentrate on low-resource languages rather than on English. Therefore, we report results of our single model. Despite that our best system reaches 69.0 $F_{0.5}$ score, which is comparable to the performance of best systems that employ ensembles. Although Grundkiewicz et al. (2019) do not report their single system score, we can hypothesise that given development set scores, our system is on par with theirs or even performs slightly better.

Note that there is a significant difference between results reported on W&I+L dev and W&I+L test sets. This is caused by the fact that each sentence in the W&I+L test set was annotated by 10 annotators, while there is only a single annotator for each sentence in the development set.

5.2 German

Boyd (2018) developed a GEC system for German based on multilayer convolutional encoder-decoder neural network (Chollampatt and Ng, 2018). To account for the lack of annotated

System	Test Subset	P	R	$F_{0.5}$
Richter et al. (2012)	All	68.72	36.75	58.54
Our work – synthetic pretrain	All	80.32	39.55	66.59
Our work – finetuned base single GPU	All	84.21	66.67	80.00
Our work – finetuned	Foreigners – Slavic	84.34	71.55	81.43
	Foreigners – Other	81.03	62.36	76.45
	Romani	86.61	71.13	83.00
	All	83.75	68.48	80.17

Table 7: Results on on AKCES-GEC Test Set (Czech).

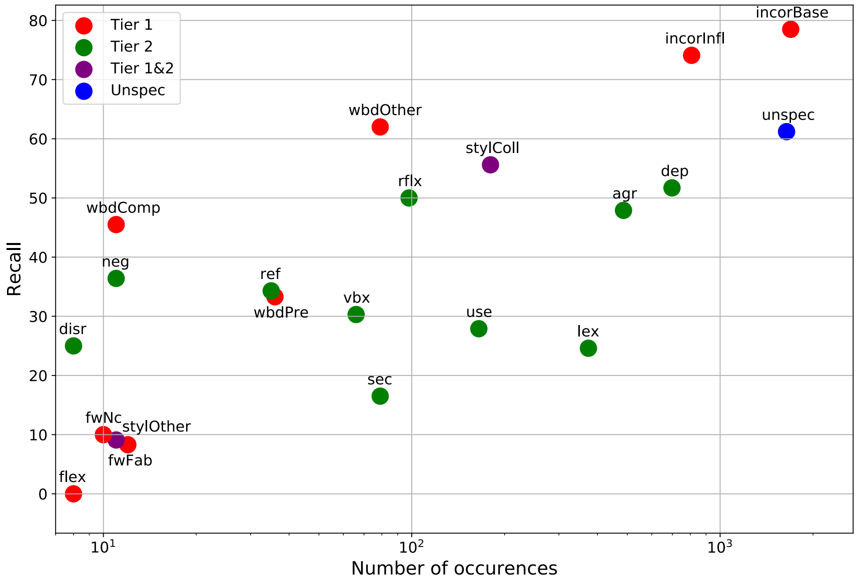

Figure 3: Recall for each error type in the test set of AKCES-GEC, computed using the first annotator (ID 0).

data, she generated additional training data from Wikipedia edits, which she filtered to match the distribution of the original error types. As Table 6 shows, her best system reaches 45.22 $F_{0.5}$ score on Falko-Merlin test set. All our three systems outperform it.

Compared to Boyd (2018), our system trained solely on synthetic data has lower recall, but substantially higher precision. The main reason behind the lower recall is the unsupervised approach to synthetic data generation. Both our finetuned models outperform Boyd (2018) system by a large margin.

5.3 Czech

We compare our system with Richter et al. (2012), who developed a statistical spelling corrector for Czech. Although their system can only make local changes (e.g., cannot insert a new word or swap two nearby words), it achieves surprisingly solid results. Nevertheless, all our three system perform better in both precision, recall and $F_{0.5}$ score. Possibly due to already quite high precision of the pretrained model, the finetuning stage improves mainly model recall.

We also evaluate performance of our best system on three subsets of the AKCES-GEC test set: Foreigners–Slavic, Foreigners–Other and Romani. As the name suggests, the first of them is a part of AKCES-GEC collected from essays of non-Czech Slavic people, the second from essays of non-Czech non-Slavic people and finally Romani comes from essays of Romani pupils with Romani ethnolect of Czech as their first language. The best result is reached on Romani subset, while on Foreigners–Other the $F_{0.5}$ score is by more than 6 points lower. We hypothesize this effect is caused by the fact, that Czech is the primary language of Romani pupils. Furthermore, we presume that foreigners with Slavic background should learn Czech faster than non-Slavic foreigners, because of the similarity between their mother tongue and

System	P	R	$F_{0.5}$
Rozovskaya and Roth (2019)	38.0	7.5	21.0
Our work – synthetic pretrain	47.76	26.08	40.96
Our work – finetuned base single GPU	59.13	26.05	47.15
Our work – finetuned	63.26	27.50	50.20

Table 8: Results on on RULEC-GEC Test Set (Russian).

Czech. This fact is supported by Table 2, which shows that the average error rate of Romani development set is 21.0%, Foreigners–Slavic 21.8% and the Foreigners–Other 23.8%.

Finally, we report recall of the best system on each error type annotated by the first annotator (ID 0) in Figure 3. Generally, our system performs better on errors annotated on Tier 1 than on errors annotated on Tier 2. Furthermore, a natural hypothesis is that the more occurrences there are for an error type, the better the recall of the system on the particular error type. Figure 3 suggests that this hypothesis seems plausible on Tier 1 errors, but its validity is unclear on Tier 2.

5.4 Russian

As Table 8 indicates, GEC in Russian currently seems to be the most challenging task. Although our system outperforms the system of Rozovskaya and Roth (2019) by more than 100% in $F_{0.5}$ score, its performance is still quite poor when compared to all previously described languages. Because the result of our system trained solely on synthetic data is comparable with the similar system for English, we hypothesise that the main reason behind these poor results is the small amount of annotated training data – while Czech has 42 210 and German 19 237 training sentence pairs, there are only 4 980 sentences in the Russian training set. To validate this hypothesis, we extended the original training set by 2 000 sentences from the development set, resulting in an increase of 3 percent points in $F_{0.5}$ score.

6 Conclusion

We presented a new dataset for grammatical error correction in Czech. It contains almost twice as much sentences as existing German dataset and more than three times as RULEC-GEC for Russian. The dataset is published in M2 format containing both separated edits and their error types.

Furthermore, we performed experiments on three low-resource languages: German, Russian and Czech. For each language, we pretrained Transformer model on synthetic data and finetuned it with a mixture of synthetic and authentic data. On all three languages, the performance of our system is substantially higher than results of the existing reported systems. Moreover, all our models supersede reported systems even if only pretrained on unsupervised synthetic data.

The performance of our system could be even higher if we trained multiple models and combined them into an ensemble. We plan to do that in future work. We also plan to extend our synthetic corpora with data modified by supervisedly extracted rules. We hope that this could help especially in case of Russian, which has the lowest amount of training data.

Acknowledgments

The work described herein has been supported by OP VVV VI LINDAT/CLARIN project (CZ.02.1.01/0.0/0.0/16_013/0001781) and it has been supported and has been using language resources developed by the LINDAT/CLARIN project (LM2015071) of the Ministry of Education, Youth and Sports of the Czech Republic. This research was also partially supported by SVV project number 260 453, GAUK 578218 of the Charles University and FP7-ICT-2010-6-257528 (MŠMT 7E11042).

References

Ondřej Bojar, Chatterjee Rajen, Christian Federmann, Yvette Graham, Barry Haddow, Huck Matthias, Koehn Philipp, Liu Qun, Logacheva Varvara, Monz Christof, et al. 2017. Findings of the 2017 conference on machine translation (wmt17). In *Second Conference onMachine Translation*, pages 169–214. The Association for Computational Linguistics.

Adriane Boyd. 2018. Using wikipedia edits in low resource grammatical error correction. In *Proceedings of the 2018 EMNLP Workshop W-NUT: The 4th Workshop on Noisy User-generated Text*, pages 79–84.

Adriane Boyd, Jirka Hana, Lionel Nicolas, Detmar Meurers, Katrin Wisniewski, Andrea Abel, Karin Schöne, Barbora Stindlová, and Chiara Vettori. 2014. The merlin corpus: Learner language and the cefr. In *LREC*, pages 1281–1288.

Christopher Bryant and Ted Briscoe. 2018. Language model based grammatical error correction without annotated training data. In *Proceedings of the Thirteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 247–253.

Christopher Bryant, Mariano Felice, Øistein E Andersen, and Ted Briscoe. 2019. The bea-2019 shared task on grammatical error correction. In *Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 52–75.

Christopher Bryant, Mariano Felice, and Edward John Briscoe. 2017. Automatic annotation and evaluation of error types for grammatical error correction. Association for Computational Linguistics.

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. *arXiv preprint arXiv:1907.01256*.

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*.

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. Association for Computational Linguistics.

Daniel Dahlmeier, Hwee Tou Ng, and Siew Mei Wu. 2013. Building a large annotated corpus of learner english: The nus corpus of learner english. In *Proceedings of the eighth workshop on innovative use of NLP for building educational applications*, pages 22–31.

Sylviane Granger. 1998. The computer learner corpus: A versatile new source of data for SLA research. In Sylviane Granger, editor, *Learner English on Computer*, pages 3–18. Addison Wesley Longman, London and New York.

Roman Grundkiewicz and Marcin Junczys-Dowmunt. 2014. The wiked error corpus: A corpus of corrective wikipedia edits and its application to grammatical error correction. In *International Conference on Natural Language Processing*, pages 478–490. Springer.

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.

Michael Heilman, Aoife Cahill, Nitin Madnani, Melissa Lopez, Matthew Mulholland, and Joel Tetreault. 2014. Predicting grammaticality on an ordinal scale. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 174–180, Baltimore, Maryland. Association for Computational Linguistics.

Tomáš Jelínek, Barbora Štindlová, Alexandr Rosen, and Jirka Hana. 2012. Combining manual and automatic annotation of a learner corpus. In *Text, Speech and Dialogue*, pages 127–134, Berlin, Heidelberg. Springer Berlin Heidelberg.

Ophélie Lacroix, Simon Flachs, and Anders Søgaard. 2019. Noisy channel for low resource grammatical error correction. In *Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 191–196.

Jared Lichtarge, Chris Alberti, Shankar Kumar, Noam Shazeer, Niki Parmar, and Simon Tong. 2019. Corpora generation for grammatical error correction. *arXiv preprint arXiv:1904.05780*.

Tomoya Mizumoto, Mamoru Komachi, Masaaki Nagata, and Yuji Matsumoto. 2011. Mining revision log of language learning sns for automated japanese error correction of second language learners. In *Proceedings of 5th International Joint Conference on Natural Language Processing*, pages 147–155.

Jakub Náplava. 2017. Natural language correction. Diploma thesis, Univerzita Karlova, Matematicko-fyzikální fakulta.

Jakub Náplava and Milan Straka. 2019. Cuni system for the building educational applications 2019 shared task: Grammatical error correction. In *Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 183–190.

Courtney Napoles, Keisuke Sakaguchi, Matt Post, and Joel Tetreault. 2015. Ground truth for grammatical error correction metrics. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 588–593.

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 the Eighteenth Conference on Computational Natural Language Learning: Shared Task*, pages 1–14.

Michal Richter, Pavel Straňák, and Alexandr Rosen. 2012. Korektor–a system for contextual spell-checking and diacritics completion. In *Proceedings of COLING 2012: Posters*, pages 1019–1028.

Alexandr Rosen. 2016. Building and using corpora of non-native Czech. In *Proceedings of the 16th ITAT: Slovenskočeský NLP workshop (SloNLP 2016)*, volume 1649 of *CEUR Workshop Proceedings*, pages 80–87, Bratislava, Slovakia. Comenius University in Bratislava, Faculty of Mathematics, Physics and Informatics, CreateSpace Independent Publishing Platform.

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.

Karel Šebesta, Zuzanna Bedřichová, Kateřina Šormová, Barbora Štindlová, Milan Hrdlička, Tereza Hrdličková, Jiří Hana, Vladimír Petkevič, Tomáš Jelínek, Svatava Škodová, Petr Janeš, Kateřina Lundáková, Hana Skoumalová, Šimon Sládek, Piotr Pierscieniak, Dagmar Toufarová, Milan Straka, Alexandr Rosen, Jakub Náplava, and Marie Poláčková. 2017. CzeSL grammatical error correction dataset (CzeSL-GEC). http://hdl.handle.net/11234/1-2143 LINDAT/CLARIN digital library at the Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University.

Rico Sennrich and Biao Zhang. 2019. Revisiting low-resource neural machine translation: A case study. *arXiv preprint arXiv:1905.11901*.

Noam Shazeer and Mitchell Stern. 2018. Adafactor: Adaptive learning rates with sublinear memory cost. *arXiv preprint arXiv:1804.04235*.

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: Short Papers-Volume 2*, pages 198–202. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Karel Šebesta. 2010. Korpusy češtiny a osvojování jazyka [Corpora of Czech and language acquisition]. In *Studie z aplikované lingvistiky [Studies in Applied Linguistics]*, volume 2010(2), pages 11–33.

Shuyao Xu, Jiehao Zhang, Jin Chen, and Long Qin. 2019. Erroneous data generation for grammatical error correction. In *Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 149–158.

Helen Yannakoudakis, Ted Briscoe, and Ben Medlock. 2011. A new dataset and method for automatically grading esol texts. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pages 180–189. Association for Computational Linguistics.

Liang-Chih Yu, Lung-Hao Lee, and Li-Ping Chang. 2014. Overview of grammatical error diagnosis for learning chinese as a foreign language. In *Proceedings of the 1st Workshop on Natural Language Processing Techniques for Educational Applications*, pages 42–47.

Zheng Yuan and Mariano Felice. 2013. Constrained grammatical error correction using statistical machine translation. In *Proceedings of the Seventeenth Conference on Computational Natural Language Learning: Shared Task*, pages 52–61.

Wajdi Zaghouani, Behrang Mohit, Nizar Habash, Ossama Obeid, Nadi Tomeh, Alla Rozovskaya, Noura Farra, Sarah Alkuhlani, and Kemal Oflazer. 2015. Large scale arabic error annotation: Guidelines and framework.

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. *arXiv preprint arXiv:1903.00138*.