Data Strategies for Low-Resource Grammatical Error Correction

Simon Flachs[2,3]*, **Felix Stahlberg**[1], and **Shankar Kumar**[1]

[1]Google Research, [2]University of Copenhagen, [3]Siteimprove
flachs@di.ku.dk,
{fstahlberg, shankarkumar}@google.com

Abstract

Grammatical Error Correction (GEC) is a task that has been extensively investigated for the English language. However, for low-resource languages the best practices for training GEC systems have not yet been systematically determined. We investigate how best to take advantage of existing data sources for improving GEC systems for languages with limited quantities of high quality training data. We show that methods for generating artificial training data for GEC can benefit from including morphological errors. We also demonstrate that noisy error correction data gathered from Wikipedia revision histories and the language learning website Lang8, are valuable data sources. Finally, we show that GEC systems pre-trained on noisy data sources can be fine-tuned effectively using small amounts of high quality, human-annotated data.

1 Introduction

Grammatical Error Correction (GEC) research has thus far been mostly focused on the English language. One reason for this narrow focus is the difficulty of the task – even for English, which has a reasonable amount of high quality data, the task is challenging. Another reason for the English-centric research has been the lack of available GEC benchmark datasets in other languages, which has made it harder to develop GEC systems on these languages.

In the past few years, there are several languages for which GEC benchmarks have become available (Davidson et al., 2020; Boyd et al., 2014; Rozovskaya and Roth, 2019; Náplava and Straka, 2019). Simultaneously, there has been considerable progress in GEC for English using cheap data sources such as artificial data and revision logs (Grundkiewicz et al., 2019; Grundkiewicz and Junczys-Dowmunt, 2014; Lichtarge et al., 2019). Since these resources are language-agnostic, the time is ripe for investigating these techniques for other languages.

Pretraining GEC systems on artificially generated errors is now common practice for English. Grundkiewicz and Junczys-Dowmunt (2019) showed good results on English, Russian, and German, using a rule-based error generation approach that Náplava and Straka (2019) extended to Czech. This approach employed the Aspell dictionary to create confusion sets of phonologically and lexically similar words. In this work, we additionally investigate the usefulness of morphology-based confusion sets. For English, model-based error generation approaches have also been shown to be useful (Kiyono et al., 2019; Stahlberg and Kumar, 2021).

State-of-the-art English GEC systems also make use of lower quality data sources, such as Wikipedia revision histories and crowd-sourced corrections from the language learning website Lang8 (Mizumoto et al., 2011; Lichtarge et al., 2019). Given that it is possible to extract data from both Wikipedia and Lang8 in multiple languages, it would be interesting to determine if this data will help improve GEC for non-English languages. Boyd (2018) have already shown promising results for German using Wikipedia revisions with a custom language-specific filtering method.

Contributions In this work we investigate data strategies for Grammatical Error Correction on languages without large quantities of high quality training data. In particular we answer the following questions: i) Can artificial error generation methods benefit from including morphological errors?; ii) How can we best make use of noisy GEC data when other data is limited?; iii) How much gold training data is necessary?

*Research conducted at Google Research.

Proceedings of the 16th Workshop on Innovative Use of NLP for Building Educational Applications, pages 117–122
April 20, 2021

	Gold			WikiEdits	Lang8
	Train	Dev	Test		
es	10,143	1,408	1,127	4,871,833	1,214,781
de	19,237	2,503	2,337	9,160,287	863,767
ru	4,980	2,500	5,000	8,482,683	684,936
cs	42,210	2,485	2,676	1,193,447	17,061

Table 1: Number of sentences for each language.

2 GEC data sources

2.1 Gold data

In recent years, high quality GEC datasets have been made available in several languages – in this work we look into Spanish (es), German (de), Russian (ru), and Czech (cs). An overview of the number of sentences for each language is shown in Table 1.

Spanish COWS-L2H (Davidson et al., 2020) is a corpus of learner Spanish corrected for grammatical errors, gathered from essays written by mostly beginner level Spanish students at the University of California at Davis.

German Falko-Merlin (Boyd, 2018) is a parallel error-correction corpus generated by merging two German learner corpora, the Falko (Reznicek et al., 2012) and Merlin (Boyd et al., 2014) corpus. The Falko part of the corpora is gathered from essays from advanced German learners, while Merlin consists of essays from a wider range of proficiency levels.

Russian RULEC-GEC (Rozovskaya and Roth, 2019) is a GEC-annotated subset of the RULEC corpus. The sources of the corpora are essays and papers written in a university setting by non-native Russian speakers of various levels.

Czech AKCES-GEC (Náplava and Straka, 2019) is a GEC corpus generated from a subset of the AKCES corpora, which consists of texts written by non-native speakers of Czech.

2.2 Artificial data

Text can be easily manipulated to destroy its grammatical structure, for example by deleting a word, or swapping the order of two words. Given that large quantities of text in multiple languages are available on the internet it is easy to produce large amounts of artificial training data. Even though these types of rule-based corruption methods do not always simulate realistic errors by human writers, they are still very useful for pre-training GEC models (Grundkiewicz et al., 2019; Grundkiewicz and Junczys-Dowmunt, 2014; Lichtarge et al., 2019).

Both rule-based and model-based methods for generating artificial data have been shown to be important components of top-performing GEC systems for English, with model-based methods currently yielding the best results (Kiyono et al., 2019). However, model-based methods typically need a large amount of training data to be able to produce an errorful data set that matches the distribution of human writers. For our low-resource setting we therefore employ a rule-based approach.

Rule-based error creation approaches using insertion, deletion and replacement operations to corrupt sentences have given good results on both English and other languages (Grundkiewicz and Junczys-Dowmunt, 2019; Náplava and Straka, 2019). Here, for word replacement operations, the Aspell dictionary is commonly used to generate confusion sets of lexically and phonetically similar words that are plausible real-world confusions (Grundkiewicz et al., 2019). Another potential source of confusion sets, which we explore in this work, is Unimorph, a database of morphological variants of words available for many languages[1] (Kirov et al., 2018).

2.3 Noisy data

Wikipedia edits Wikipedia is a publicly available, online encyclopedia for which all content is communally created and curated, and is currently available for 316 languages.[2] Wikipedia maintains a revision history of each page, making it possible to extract edits made between subsequent revisions. A subset of the edits contain corrections for grammatical errors. However there are many other types of edits unrelated to the GEC task, such as stylistic changes, vandalism etc. This noise poses a challenge for training GEC systems.

Wikipedia edit history is commonly used for training English GEC systems (Grundkiewicz and Junczys-Dowmunt, 2014; Lichtarge et al., 2019), and has also been shown useful for German, when using a custom language-specific filtering method (Boyd, 2018). To keep our experiments language-independent, we do not use this filtering method. Instead, we expect that the effects of noise in the Wikipedia data would be mitigated by subsequent

[1] http://unimorph.org

[2] https://meta.wikimedia.org/wiki/List_of_Wikipedias

finetuning on gold data. For our experiments, we use the data generation scripts from Lichtarge et al. (2019) to gather training examples from the Wikipedia edit history (see Table 1); we refer to this data source as WIKIEDITS.

Lang8 Lang8 is a social language learning website, where users can post texts in a language they are learning, which are then corrected by other users who are native or proficient speakers of the language. The website contains relatively large quantities of sentences with their corrections (Table 1) which can be used for training GEC models (Mizumoto et al., 2011). Lang8, however, also contains considerable noise. The corrections may include additional comments. Also, there is high variability in the language proficiency of users providing the corrections.

3 Systems

For all experiments we use the *Transformer* sequence-to-sequence model (Kiyono et al., 2019) available in the Tensor2tensor library.[3] The model is trained with early stopping, using Adafactor as optimizer with inverse square root decay (Shazeer and Stern, 2018). A detailed overview of hyperparameters is listed in Appendix A.[4]

We compare our results to two baseline GEC systems, Grundkiewicz and Junczys-Dowmunt (2019) (G&J) and Náplava and Straka (2019) (N&S), which have both been evaluated on Russian and German, and for Náplava and Straka (2019) additionally on Czech. Both of these systems are pretrained on artificial data and finetuned on gold data. When training the models several strategies were used: source and target word dropouts, edit-weighted maximum likelihood estimation and checkpoint averaging. In this work we do not employ these techniques because our focus is on comparing methods for data collection and generation and less on surpassing the state-of-the-art.

4 Experiments

We evaluate our models using $F_{0.5}$ score computed using the MaxMatch scorer (Dahlmeier and Ng, 2012). For all experiments, the reported scores are computed for the model trained on the specified data source, further finetuned on the gold training data.

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

[4] We used the "transformer_clean_big_tpu" setting

	cs	de	ru	es
		Artificial		
Unimorph	71.08	60.87	32.91	44.68
Aspell	71.53	**63.49**	32.86	**48.22**
Aspell+Unimorph	**71.90**	62.55	**35.95**	48.20
		WikiEdits		
WikiEdits	55.14	58.00	23.92	47.35
Artificial→WikiEdits	**74.64**	**66.74**	40.68	**52.56**
Artificial+WikiEdits	72.91	66.66	**42.80**	51.55
		Summary		
N&S (2019)	**80.17**	**73.71**	**50.20**	-
G&J (2019)	-	70.24	34.46	-
Artificial	71.90	63.49	35.95	48.22
+ WikiEdits	74.64	66.74	42.80	52.56
+ Lang8	75.07	69.24	44.72	**57.32**

Table 2: $F_{0.5}$ scores of experiments on the ARTIFICIAL, WIKIEDITS, and LANG8 data sources.

4.1 Creating artificial data

We first investigate if artificial data creation methods can benefit from the inclusion of morphology-based confusion sets generated from Unimorph.

We train the systems on 10 million examples generated from the WMT News Crawl using the rule-based method from Náplava and Straka (2019) which is a modification of the method presented by Grundkiewicz and Junczys-Dowmunt (2019).

First, for each sentence a word-level (or character-level) error probability is sampled from a normal distribution with a predefined mean and standard deviation. The number of words (or characters) to corrupt are then decided by multiplying the probability by the number of words (or characters) in the sentence. Each corruption is then performed using one of the following operations: insert, swap-right, substitute and delete. Furthermore, at the word level an operation to change the casing is included and at the character level an operation to replace diacritics is included. The operation to apply is selected based on probabilities estimated from the development sets. All parameters used in our experiments are presented in Appendix B.

When creating the artificial data we report three experiments – for the word substitution operation a replacement word is chosen from a confusion set generated by either 1) Aspell; 2) Unimorph; or 3) Aspell or Unimorph with equal likelihood (Aspell + Unimorph).

Table 2 shows that using only Unimorph performs the worst. This is expected since the system

would only learn to correct morphological substitution errors. Mixing Aspell and Unimorph works better for Russian and Czech but for the other languages, using Aspell alone performs better. Thus including Unimorph can help for morphological rich languages, such as Russian and Czech. We will refer to the best performing artificially created dataset for each language as ARTIFICIAL.

4.2 Including noisy data

We next investigate whether data extracted from Wikipedia revisions and Lang8 can improve our systems even further.

WIKIEDITS We perform three experiments: 1) training on WIKIEDITS from scratch; 2) fine-tuning on WIKIEDITS, starting from models pre-trained on ARTIFICIAL (ARTIFICIAL→WIKIEDITS); and 3) training on an equal-proportion mix of ARTIFICIAL and WIKIEDITS (ARTIFICIAL + WIKIEDITS). From Table 2, training only on WIKIEDITS performs worse than the models trained solely on ARTIFICIAL. However, finetuning the ARTIFICIAL-trained model on WIKIEDITS gives a large improvement. This suggests that the model primed for the GEC task by pre-training on ARTIFICIAL can better handle the noise in WIKIEDITS. Mixing the two sources is generally worse, indicating that WIKIEDITS, despite its noise, is of a higher quality and contains realistic GEC errors. However, this is not the case for Russian, where it is better to mix the two data sources. This suggests that Russian Wikipedia revisions are more likely to be unrelated to GEC, and mixing it with ARTIFICIAL regularizes this noise.

LANG8 Fine-tuning the best model from the WIKIEDITS experiments on LANG8 improves performance on all languages (Table 2), which confirms the utility of this data source as a valuable source of grammatical corrections.

4.3 How much gold data do we need?

Human annotated (Gold) data is a scarce resource, as human annotations are expensive. Therefore it is important to determine how much data is necessary to train useful GEC systems in new languages. We analyze the performance of systems finetuned on increasingly larger subsets of available data.

We investigate two scenarios: 1) finetuning a model pretrained only on ARTIFICIAL, and 2) finetuning a model pretrained on ARTIFICIAL, WIKIEDITS, and LANG8 (using the best method from previous experiments). This ablation allows us to assess whether noisy data sources can ameliorate the need for gold data.

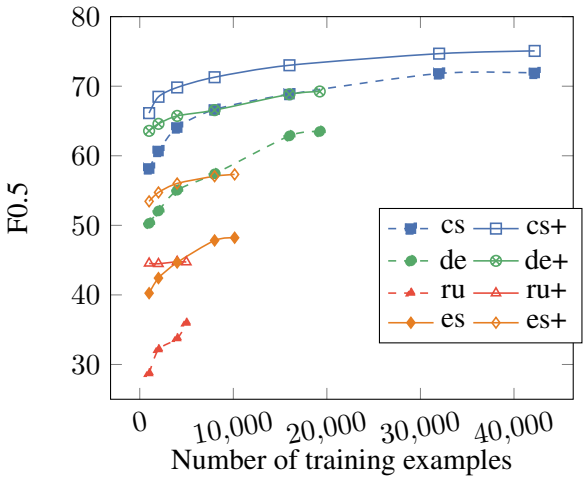

Figure 1: GEC performance ($F_{0.5}$) for different amounts of gold training data. Systems have been pretrained on ARTIFICIAL. The + denotes system has additionally been pretrained on WIKIEDITS and LANG8

Performance curves (Figure 1) flatten out quickly at around 15k sentences, suggesting that not much data is needed. This is especially the case when the system has additionally been trained on WIKIEDITS and LANG8. This demonstrates that it is possible to obtain a reasonable quality without much human-annotated data in new languages.

5 Conclusion

In this paper we have investigated how best to make use of available data sources for GEC in low resource scenarios. We have shown a set of best practices for using artificial data, Wikipedia revision data and Lang8 data, that gives good results across four languages.

We show that using Unimorph for generating artificial data is useful for Russian and Czech, which are morphologically rich languages. Using Wikipedia edits is a valuable source of data, despite its noise. Lang8 is an even better source of high-quality GEC data, despite its smaller size and uncertainties associated with crowdsourcing. When using gold data for fine-tuning, even small amounts of data can yield good results. This is especially true when the initial model has been pretrained on Wikipedia edits and Lang8. We expect this work to provide a good starting point for developing GEC systems for a wider range of languages.

References

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, Brussels, Belgium. Association for Computational Linguistics.

Adriane Boyd, Jirka Hana, Lionel Nicolas, Detmar Meurers, Katrin Wisniewski, Andrea Abel, Karin Schöne, Barbora Štindlová, and Chiara Vettori. 2014. The MERLIN corpus: Learner language and the CEFR. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 1281–1288, Reykjavik, Iceland. European Language Resources Association (ELRA).

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

Sam Davidson, Aaron Yamada, Paloma Fernandez Mira, Agustina Carando, Claudia H. Sanchez Gutierrez, and Kenji Sagae. 2020. Developing NLP tools with a new corpus of learner Spanish. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 7238–7243, Marseille, France. European Language Resources Association.

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 and Marcin Junczys-Dowmunt. 2019. Minimally-augmented grammatical error correction. In *Proceedings of the 5th Workshop on Noisy User-generated Text (W-NUT 2019)*, pages 357–363, Hong Kong, China. Association for Computational Linguistics.

Roman Grundkiewicz, Marcin Junczys-Dowmunt, and Kenneth Heafield. 2019. Neural grammatical error correction systems with unsupervised pre-training on synthetic data. In *Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 252–263, Florence, Italy. Association for Computational Linguistics.

Christo Kirov, Ryan Cotterell, John Sylak-Glassman, Géraldine Walther, Ekaterina Vylomova, Patrick Xia, Manaal Faruqui, Sebastian Mielke, Arya McCarthy, Sandra Kübler, David Yarowsky, Jason Eisner, and Mans Hulden. 2018. UniMorph 2.0: Universal morphology. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC-2018)*, Miyazaki, Japan. European Languages Resources Association (ELRA).

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.

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.

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, Chiang Mai, Thailand. Asian Federation of Natural Language Processing.

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.

Marc Reznicek, Anke Lüdeling, Cedric Krummes, Franziska Schwantuschke, Maik Walter, Karin Schmidt, Hagen Hirschmann, and Torsten Andreas. 2012. Das falko-handbuch. korpusaufbau und annotationen version 2.01.

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.

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

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

A Model hyperparameters

An overview of model hyperparameters used for our GEC system:

- 6 layers for both the encoder and the decoder.
- 8 attention heads.
- A dictionary of 32k word pieces.
- Embedding size $d_{model} = 1024$.
- Position-wise feed forward network at every layer of inner size d_{ff} = 4096.
- Batch size = 4096.
- For inference we use beam search with a beam width of 4.
- When pretraining we set the learning rate to 0.2 for the first 8000 steps, then decrease it proportionally to the inverse square root of the number of steps after that.
- When finetuning, we use a constant learning rate of 3×10^{-5}.

B Artificial data parameters

Language	**Token-level operations**					**Character-level operations**				
	sub	**ins**	**del**	**swap**	**recase**	**sub**	**ins**	**del**	**swap**	**toggle diacritics**
es	0.69	0.17	0.11	0.01	0.02	0.25	0.25	0.25	0.25	0
cs	0.7	0.1	0.05	0.1	0.05	0.2	0.2	0.2	0.2	0.2
de	0.64	0.2	0.1	0.01	0.05	0.25	0.25	0.25	0.25	0
ru	0.65	0.1	0.1	0.1	0.05	0.25	0.25	0.25	0.25	0

Table 3: Language specific parameters for token- and character-level noising operations. For all languages word error rate is set to 0.15 and character error rate to 0.02