Using Wikipedia Edits in Low Resource Grammatical Error Correction

Adriane Boyd
Department of Linguistics
University of Tübingen
adriane@sfs.uni-tuebingen.de

Abstract

We develop a grammatical error correction (GEC) system for German using a small gold GEC corpus augmented with edits extracted from Wikipedia revision history. We extend the automatic error annotation tool ERRANT (Bryant et al., 2017) for German and use it to analyze both gold GEC corrections and Wikipedia edits (Grundkiewicz and Junczys-Dowmunt, 2014) in order to select as additional training data Wikipedia edits containing grammatical corrections similar to those in the gold corpus. Using a multilayer convolutional encoder-decoder neural network GEC approach (Chollampatt and Ng, 2018), we evaluate the contribution of Wikipedia edits and find that carefully selected Wikipedia edits increase performance by over 5%.

1 Introduction and Previous Work

In the past decade, there has been a great deal of research on grammatical error correction for English including a series of shared tasks, Helping Our Own in 2011 and 2012 (Dale and Kilgarriff, 2011; Dale et al., 2012) and the CoNLL 2013 and 2014 shared tasks (Ng et al., 2013, 2014), which have contributed to the development of larger English GEC corpora. On the basis of these resources along with advances in machine translation, the current state-of-the-art English GEC systems use ensembles of neural MT models (Chollampatt and Ng, 2018) and hybrid systems with both statistical and neural MT models (Grundkiewicz and Junczys-Dowmunt, 2018).

In addition to using gold GEC corpora, which are typically fairly small in the context of MT-based approaches, research in GEC has taken a number of alternate data sources into consideration such as artificially generated errors (e.g., Wagner et al., 2007; Foster and Andersen, 2009; Yuan and Felice, 2013), crowd-sourced corrections (e.g., Mizumoto et al., 2012), or errors from native language resources (e.g., Cahill et al., 2013; Grundkiewicz and Junczys-Dowmunt, 2014). For English, Grundkiewicz and Junczys-Dowmunt (2014) extracted pairs of edited sentences from the Wikipedia revision history and filtered them based on a profile of gold GEC data in order to extend the training data for a statistical MT GEC system and found that the addition of filtered edits improved the system's $F_{0.5}$ score by ~2%. For languages with more limited resources, native language resources such as Wikipedia offer an easily accessible source of additional data.

Using a similar approach that extends existing gold GEC data with Wikipedia edits, we develop a neural machine translation grammatical error correction system for a new language, in this instance German, for which there are only small gold GEC corpora but plentiful native language resources.

2 Data and Resources

The following sections describe the data and resources used in our experiments on GEC for German. We create a new GEC corpus for German along with the models needed for the neural GEC approach presented in Chollampatt and Ng (2018). Throughout this paper we will refer to the source sentence as the *original* and the target sentence as the *correction*.

2.1 Gold GEC Corpus

As we are not aware of any standard corpora for German GEC, we create a new grammatical error correction corpus from two German learner corpora that have been manually annotated following similar guidelines. In the Falko project, annotation guidelines were developed for *minimal target hypotheses*, minimal corrections that transform an original sentence into a grammatical correction, and these guidelines were applied to ad-

Proceedings of the 2018 EMNLP Workshop W-NUT: The 4th Workshop on Noisy User-generated Text, pages 79–84
Brussels, Belgium, Nov 1, 2018.

Corpus		# Sent	Err/S	Err/Tok
Falko	Train	11038	2.90	0.15
	Dev	1307	2.87	0.16
	Test	1237	3.00	0.16
MERLIN	Train	9199	2.63	0.20
	Dev	1196	2.65	0.20
	Test	1100	2.54	0.21
Total		24077	2.77	0.18

Table 1: Falko-MERLIN German GEC Corpus

vanced German learner essays (Reznicek et al., 2012). The MERLIN project (Boyd et al., 2014) adapted the Falko guidelines and annotated learner texts from a wide range of proficiency levels.[1]

We extract pairs of original sentences and corrections from all annotated sentence spans in FalkoEssayL2 v2.4[2] (248 texts), FalkoEssayWhig v2.0[2] (196 texts), and MERLIN v1.1[3] (1,033 texts) to create the new Falko-MERLIN GEC Corpus, which contains 24,077 sentence pairs. The corpus is divided into train (80%), dev (10%), and test (10%) sets, keeping all sentences from a single learner text within the same partition.

An overview of the Falko-MERLIN GEC Corpus is shown in Table 1 with the number of errors per sentence and errors per token as analyzed by ERRANT for German (see section 3.1). On average, the Falko corpus (advanced learners) contains longer sentences with fewer errors per token while the MERLIN corpus (all proficiency levels) contains shorter sentences with more errors per token. A more detailed ERRANT-based analysis is presented in Figure 2 in section 3.2.

2.2 Wikipedia

In our experiments, we use German Wikipedia dumps of articles and revision history from June 1, 2018. Wikipedia edits are extracted from the revision history using Wiki Edits (Grundkiewicz and Junczys-Dowmunt, 2014) with a maximum sentence length of 60 tokens, since 99% of the Falko and MERLIN sentences are shorter than 60 tokens. For training the subword embeddings, plain text is extracted from the German Wikipedia articles using WikiExtractor.[4]

[1] We also considered including German data from Lang-8, however it seemed to be far too noisy.

[2] https://www.linguistik.hu-berlin.de/de/institut/professuren/korpuslinguistik/forschung/falko/zugang

[3] https://www.merlin-platform.eu

[4] https://github.com/attardi/wikiextractor

2.3 BPE Model and Subword Embeddings

We learn a byte pair encoding (BPE) (Sennrich et al., 2016) with 30K symbols using the corrections from the Falko-MERLIN training data plus the complete plain Wikipedia article text. As suggested by Chollampatt and Ng (2018), we encode the Wikipedia article text using the BPE model and learn fastText embeddings (Bojanowski et al., 2017) with 500 dimensions.

2.4 Language Model

For reranking, we train a language model on the first one billion lines (~12 billion tokens) of the deduplicated German Common Crawl corpus (Buck et al., 2014).

3 Method

We extend the Falko-MERLIN GEC training data with sentence-level Wikipedia edits that include similar types of corrections. In order to automatically analyze German GEC data, we extend ERRANT from English to German (section 3.1) and use its analyses to select suitable Wikipedia edits (section 3.2).

3.1 ERRANT

ERRANT, the ERRor ANnotation Tool (Felice et al., 2016; Bryant et al., 2017), analyzes pairs of English sentences from a GEC corpus to identify the types of corrections performed. The tokens in a pair of sentences are aligned using Damerau-Levenshtein edit distance with a custom substitution cost that includes linguistic information — lemmas, POS, and characters — to promote alignments between related word forms. After the individual tokens are aligned, neighboring edits are evaluated to determine whether two or more edits should be merged into one longer edit, such as merging *wide* → *widespread* followed by *spread* → ∅ into a single edit *wide spread* → *widespread*.

To assign an error type to a correction, ERRANT uses a rule-based approach that considers information about the POS tags, lemmas, stems, and dependency parses. To extend ERRANT for German, we adapted and simplified the English error types, relying on UD POS tags instead of language-specific tags as much as possible. Our top-level German ERRANT error types are shown with examples in Table 2. For substitution errors,

Error Type	Example
POS (15)	*dem* → *den* (DET:FORM)
MORPH	*solid* → *solide*
ORTH	*Große* → *große*
SPELL	*wächseln* → *wechseln*
ORDER	*zu gehen* → *gehen zu*
CONTR	*'s* → ∅
OTHER	*hochem* → *einem hohen*

Table 2: German ERRANT Error Types

each POS error type has an additional FORM subtype if the tokens have the same lemma.

The POS tag types include 14 UD POS types plus the German-specific STTS tag TRUNC. The MORPH tag captures errors for related word forms with different POS tags, ORTH is for capitalization and whitespace errors, SPELL errors have an original token that is not in a large word list with >50% overlapping characters compared to the corrected token, ORDER errors cover adjacent reordered tokens, and CONTR errors involve the contraction *'s* ('it'). All remaining errors are classified as OTHER.

In ERRANT for English, all linguistic annotation is performed with spaCy.[5] We preserve as much of the spaCy pipeline as possible using spaCy's German models, however the lemmatizer is not sufficient and is replaced with the TreeTagger lemmatizer.[6] All our experiments are performed with spaCy 2.0.11 and spaCy's default German model. The word list for detecting spelling errors comes from Hunspell igerman98-20161207[7] and the mapping of STTS to UD tags from TuebaUDConverter (Çöltekin et al., 2017).

An example of a German ERRANT analysis is shown in Figure 1. The first token is analyzed as an adjective substitution error where both adjectives have the same lemma (S:ADJ:FORM), the inflected deverbal adjective *bestandenen* 'passed' is inserted before *Prüfung* 'exam' (I:ADJ), and the past participle *bestanden* 'passed' is deleted at the end of the sentence (D:VERB). Note that ERRANT does not analyze *Prüfung bestanden* → *bestandenen Prüfung* as a word order error because the reordered word forms are not identical. In cases like these and ones with longer distance movement, which is a frequent type of correction in non-native German texts, ERRANT has no way to indicate that these two word forms are related or that this pair of edits is coupled.

[5] https://spacy.io

[6] http://www.cis.uni-muenchen.de/~schmid/tools/TreeTagger/

[7] https://www.j3e.de/ispell/igerman98/dict/

3.2 Filtering Edits with ERRANT

Even though the Wiki Edits algorithm (Grundkiewicz and Junczys-Dowmunt, 2014) extracts only sentence pairs with small differences, many edits relate to content rather than grammatical errors, such as inserting a person's middle name or updating a date. In order to identify the most relevant Wikipedia edits for GEC, we analyze the gold GEC corpus and Wikipedia edits with ERRANT and then filter the Wikipedia edits based on a profile of the gold GEC data.

First, sentences with ERRANT error types that indicate content or punctuation edits are discarded: 1) sentences with only punctuation, proper noun, and/or OTHER error types, 2) sentences with edits modifying only numbers or non-Latin characters, and 3) sentences with OTHER edits longer than two tokens. Second, the ERRANT profile of the gold corpus is used to select edits that: 1) include an original token edited in the gold corpus, 2) include the same list of error types as a sentence in the gold corpus, 3) include the same set of error types as a sentence in the gold corpus for 2+ error types, or 4) for sets of $Gold$ and $Wiki$ error types have a Jaccard similarity coefficient to a gold sentence greater than 0.5:

$$J(Gold, Wiki) = \frac{|Gold \cap Wiki|}{|Gold \cup Wiki|}$$

After ERRANT-based filtering, approximately one third of the sentences extracted with Wiki Edits remain.

The distribution of selected ERRANT error types for the Falko and MERLIN gold GEC corpora vs. the unfiltered and filtered Wikipedia edit corpora are shown in Figure 2 in order to provide an overview of the similarities and differences between the data. As intended, filtering Wikipedia edits as described above decreases the number of potentially content-related PNOUN and OTHER edits while increasing the proportion of other types of edits. Both in the unfiltered and filtered Wikipedia edits corpora, the overall frequency of errors remains lower than in the Falko-MERLIN GEC corpus: 1.7 vs. 2.8 errors per sentence and 0.08 vs. 0.18 errors per token.

Original	Herzliche	Glückwunsch	zur		Prüfung	bestanden	.
Correction	Herzlichen	Glückwunsch	zur	bestandenen		Prüfung	.
ERRANT	S:ADJ:FORM			I:ADJ		D:VERB	
	heartfelt	congratulation	to the	passed	exam		.

'Congratulations on passing your exam.'

Figure 1: Example German ERRANT Analysis

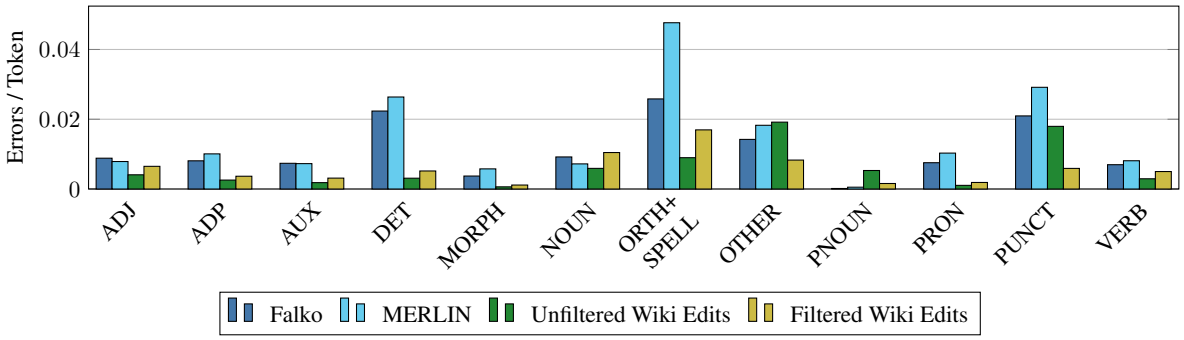

Figure 2: Distribution of Selected ERRANT Error Types

Training Data	**Unfiltered Wiki Edits**			**Filtered Wiki Edits**		
	P	R	$F_{0.5}$	P	R	$F_{0.5}$
Falko-MERLIN (19K)	45.38	25.42	39.22	45.38	25.42	39.22
+ 100K Wiki Edits	53.91	22.44	42.10	54.59	22.25	42.30
+ 250K Wiki Edits	57.57	21.80	**43.35**	57.30	23.04	44.17
+ 500K Wiki Edits	**58.55**	20.33	42.55	58.74	22.37	44.33
+ 1M Wiki Edits	57.86	21.72	43.41	**60.19**	21.75	44.47
+ 1M Wiki Edits + EO	41.43	**28.74**	38.07	39.95	29.03	37.15
+ 1M Wiki Edits + LM	44.72	28.39	40.11	51.81	29.26	44.89
+ 1M Wiki Edits + LM_{Norm}	48.65	28.69	42.71	51.99	**29.73**	**45.22**
1M Wiki Edits Only	31.12	5.33	15.82	30.13	5.42	15.75
1M Wiki Edits Only + EO	19.66	11.40	17.17	20.26	12.18	17.89
1M Wiki Edits Only + LM	26.34	12.59	21.62	29.12	13.95	23.92
1M Wiki Edits Only + LM_{Norm}	25.21	12.38	20.88	29.96	13.95	24.37

Table 3: Results for MLConv GEC on Falko-Merlin Test Set (M^2)

4 Results and Discussion

We evaluate the effect of extending the Falko-MERLIN GEC Corpus with Wikipedia edits for a German GEC system using the multilayer convolutional encoder-decoder neural network approach from Chollampatt and Ng (2018), using the same parameters as for English.[8] We train a single model for each condition and evaluate on the Falko-MERLIN test set using M^2 scorer (Dahlmeier and Ng, 2012).[9]

[8] https://github.com/nusnlp/mlconvgec2018

[9] https://github.com/nusnlp/m2scorer/archive/version3.2.tar.gz

The results, presented in Table 3, show that the addition of both unfiltered and filtered Wikipedia edits to the Falko-MERLIN GEC training data lead to improvements in performance, however larger numbers of unfiltered edits (>250K) do not consistently lead to improvements, similar to the results for English in Grundkiewicz and Junczys-Dowmunt (2014). However for filtered edits, increasing the number of additional edits from 100K to 1M continues to lead to improvements, with an overall improvement of 5.2 $F_{0.5}$ for 1M edits over the baseline without additional reranking.

In contrast to the results for English in Chollampatt and Ng (2018), edit operation (EO) rerank-

ing decreases scores in conditions with gold GEC training data in our experiments and reranking with a web-scale language model (LM) does not consistently increase scores, although both reranking methods lead to increases in recall. The best result of 45.22 $F_{0.5}$ is obtained with Falko-MERLIN + 1M Filtered Wiki Edits with language model reranking that normalizes scores by the length of the sentence.

An analysis of the performance on Falko vs. MERLIN shows stronger results for MERLIN, with 44.19 vs. 46.52 $F_{0.5}$ for Falko-MERLIN + 1M Filtered Wiki Edits + LM_{Norm}. We expected the advanced Falko essays to benefit from being more similar to Wikipedia than MERLIN, however MERLIN may simply contain more spelling and inflection errors that are easy to correct given a small amount of context.

In order to explore the possibility of developing GEC systems for languages with fewer resources, we trained models solely on Wikipedia edits, which leads to a huge drop in performance (45.22 vs. 24.37 $F_{0.5}$). However, the genre differences may be too large to draw solid conclusions and this approach may benefit from further work on Wikipedia edit selection, such as using a language model to exclude some Wikipedia edits that introduce (rather than correct) grammatical errors.

5 Future Work

The combined basis of ERRANT and Wiki Edits make it possible to explore MT-based GEC approaches for languages with limited gold GEC resources. The current German ERRANT error analysis approach can be easily generalized to rely on a pure UD analysis, which would make it possible to apply ERRANT to any language with a UD parser and a lemmatizer. Similarly, the process of filtering Wikipedia edits could use alternate methods in place of a gold reference corpus, such as a list of targeted token or error types, to generate GEC training data for any language with resources similar to a Wikipedia revision history.

For the current German GEC system, a detailed error analysis for the output could identify the types of errors where Wikipedia edits make a significant contribution and other areas where additional data could be incorporated, potentially through artificial error generation or crowd-sourcing.

6 Conclusion

We provide initial results for grammatical error correction for German using data from the Falko and MERLIN corpora augmented with Wikipedia edits that have been filtered using a new German extension of the automatic error annotation tool ERRANT (Bryant et al., 2017). Wikipedia edits are extracted using Wiki Edits (Grundkiewicz and Junczys-Dowmunt, 2014), profiled with ERRANT, and filtered with reference to the gold GEC data. We evaluate our method using the multi-layer convolutional encoder-decoder neural network GEC approach from Chollampatt and Ng (2018) and find that augmenting a small gold German GEC corpus with one million filtered Wikipedia edits improves the performance from 39.22 to 44.47 $F_{0.5}$ and additional language model reranking increases performance to 45.22. The data and source code for this paper are available at: https://github.com/adrianeboyd/boyd-wnut2018/

Acknowledgments

We are grateful to the anonymous reviewers for their helpful feedback. This work was supported by the German Research Foundation (DFG) under project ME 1447/2-1.

References

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

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 LREC 2014*, Reykjavik, Iceland. European Language Resources Association (ELRA).

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

Christian Buck, Kenneth Heafield, and Bas van Ooyen. 2014. N-gram counts and language models from the Common Crawl. In *Proceedings of the Language Resources and Evaluation Conference*, Reykjavik, Iceland.

Aoife Cahill, Nitin Madnani, Joel Tetreault, and Diane Napolitano. 2013. Robust systems for preposition error correction using Wikipedia revisions. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 507–517. Association for Computational Linguistics.

Çağrı Çöltekin, Ben Campbell, Erhard Hinrichs, and Heike Telljohann. 2017. Converting the TüBa-D/Z treebank of german to Universal Dependencies. In *Proceedings of the NoDaLiDa 2017 Workshop on Universal Dependencies (UDW 2017)*, pages 27–37, Gothenburg, Sweden.

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.

Robert Dale, Ilya Anisimoff, and George Narroway. 2012. A report on the preposition and determiner error correction shared task. In *Proceedings of the 7th Workshop on the Innovative Use of NLP for Building Educational Applications*, pages 54–62.

Robert Dale and Adam Kilgarriff. 2011. Helping Our Own: The HOO 2011 pilot shared task. In *Proceedings of the Generation Challenges Session at the 13th European Workshop on Natural Language Generation*, pages 242–249, Nancy, France. Association for Computational Linguistics.

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

Jennifer Foster and Øistein Andersen. 2009. Generrate: Generating errors for use in grammatical error detection. In *Proceedings of the Fourth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 82–90. Association for Computational Linguistics.

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 *Advances in Natural Language Processing – Lecture Notes in Computer Science*, volume 8686, pages 478–490. Springer.

Roman Grundkiewicz and Marcin Junczys-Dowmunt. 2018. Near human-level performance in grammatical error correction with hybrid machine translation. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 284–290. Association for Computational Linguistics.

Tomoya Mizumoto, Yuta Hayashibe, Mamoru Komachi, Masaaki Nagata, and Yuji Matsumoto. 2012. The effect of learner corpus size in grammatical error correction of ESL writings. In *Proceedings of COLING 2012: Posters*, pages 863–872. The COLING 2012 Organizing Committee.

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. Association for Computational Linguistics.

Hwee Tou Ng, Siew Mei Wu, Yuanbin Wu, Christian Hadiwinoto, and Joel Tetreault. 2013. The CoNLL-2013 shared task on grammatical error correction. In *Proceedings of the Seventeenth Conference on Computational Natural Language Learning: Shared Task*, pages 1–12. Association for Computational Linguistics.

Marc Reznicek, Anke Lüdeling, Cedric Krummes, and Franziska Schwantuschke. 2012. *Das Falko-Handbuch. Korpusaufbau und Annotationen Version 2.0*.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725. Association for Computational Linguistics.

Joachim Wagner, Jennifer Foster, and Josef van Genabith. 2007. A comparative evaluation of deep and shallow approaches to the automatic detection of common grammatical errors. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 112–121, Prague, Czech Republic. Association for Computational Linguistics.

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. Association for Computational Linguistics.