IMPARA: Impact-based Metric for GEC using Parallel Data

Anonymous ACL submission

Abstract

Automatic evaluation of Grammatical Error Correction (GEC) is essential in developing efficient GEC systems. Existing methods for automatic evaluation require multiple reference sentences or manual scores. However, such resources are costly, which hinders automatic evaluation for various domains and correction types. This paper proposes IMpact-based metric for GEC using PARAllel data (**IMPARA**) that utilizes parallel data consisting of pairs of grammatical/ungrammatical sentences and correction impacts. Because parallel data can be obtained with less effort than manually assessing evaluation scores, IMPARA can reduce the cost of data creation. Correlations between IMPARA and human scores show that IMPARA is comparable or better than existing methods. Furthermore, we find that IMPARA can perform evaluations that fit different domains and correction styles by changing the parallel data.

1 Introduction

GEC is the task of correcting grammatically incorrect sentences (Yuan and Briscoe, 2016; Chollampatt and Ng, 2018; Junczys-Dowmunt et al., 2018; Kaneko et al., 2020; Omelianchuk et al., 2020). GEC is useful in various domains including website text (Flachs et al., 2020) and essays written by language learners (Yannakoudakis et al., 2011). Moreover, GEC systems have different correction styles such as minimal and fluency edits (Ng et al., 2013; Napoles et al., 2017; Hotate et al., 2019). A GEC model is evaluated by computing correlations between automatic and manual corrections. Because the cost of a manual evaluation is high, we need to establish an automatic evaluation measures that correlate well with manual evaluation.

Automatic evaluation measures of GEC are categorized into two. One is reference-based methods (Dahlmeier and Ng, 2012; Napoles et al., 2015; Bryant et al., 2017) that evaluate the closeness of output sentences from a GEC system and the reference sentences created by human annotators. In general, an ungrammartical sentence can be corrected in different ways. Therefore, reference-based methods require multiple reference sentences for accurate evaluation. However, Choshen and Abend (2018b) argue that it is unrealistic to prepare sufficient reference sentences that cover all correction patterns. In addition, they show that using low-coverage reference sets deteriorates the reliability of reference-based evaluation.

The other category includes reference-less methods that use only input sentences and system outputs. Researchers proposed several reference-less methods based on language models (Napoles et al., 2016; Flachs et al., 2020). However, they do not leverage GEC specific supervision data, which causes low correlations with manual evaluation. Therefore, Asano et al. (2017) and Yoshimura et al. (2020) proposed reference-less methods optimized directly for manual evaluation. These methods require manual evaluation to adapt evaluation models for different domains and correction styles. Still, it is difficult and costly to create a reliable data for manual evaluation (Choshen and Abend, 2018a).

In order to realize an accurate evaluation metric at a lower cost, we propose a reference-less method **IMPARA**[1] that can be trained only on parallel data consisting of grammatical and ungrammatical sentence pairs. We introduce the idea of correction impact to effectively train an evaluation model from parallel data. IMPARA can use parallel data in the same format as GEC training data, which greatly reduces the cost of data creation. In addition, an IMPARA model can take into account the characteristics of various domains and correction styles.

Meta-evaluation experiments show that IMPARA has the comparable or better evaluation performance than existing reference-less methods (Yoshimura et al., 2020; Flachs et al., 2020). Fur-

[1] `https://...` (see the attached code during the review)

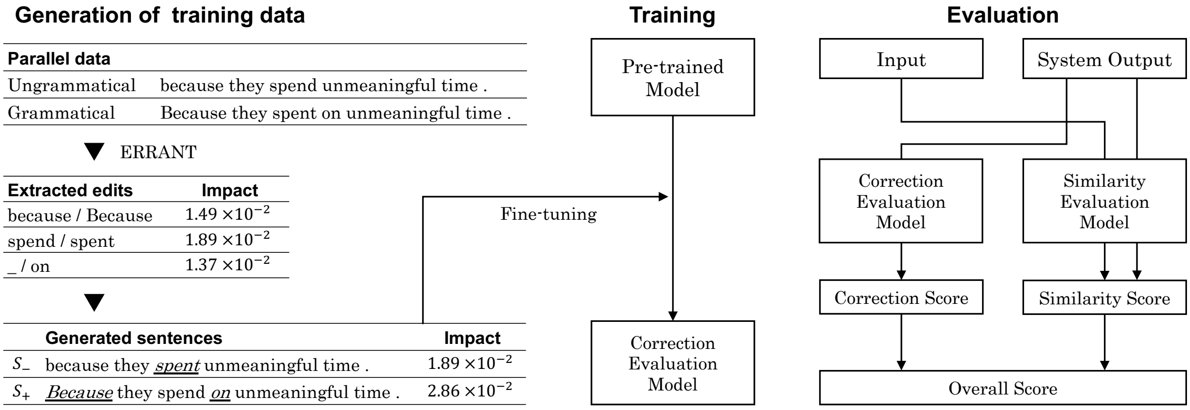

Figure 1: Generation of supervision data (left), training (middle), and the usage (right) of an IMPARA model.

thermore, we find that training an IMPARA model on data from the domain and correction style corresponding to the meta-evaluation data improves evaluation performance.

2 IMPARA

Figure 1 illustrates IMPARA, which consists of two evaluation models of correction and similarity.

The Correction Evaluation (CE) model computes a relative correction score to an output sentence. The CE model was inspired by MAEGE (Choshen and Abend, 2018a), which meta-evaluates the automatic evaluation measure using sentence pairs ranked by the number of editing operations applied to ungrammatical sentences. The model learns order relation by comparing edited sentence pairs where partial edits are applied at random to an ungrammatical sentence. We assume that each edit corrects an error of a different severity. Therefore, we introduce the *impact* of an edit and detemine an order relation on edited sentence pairs.

The Similarity Evaluation (SE) model prevents deviations of an output sentence from an input. While Islam and Magnani (2021) computes the similarity score between input and output sentences at surface level, the proposed SE model computes a similarity score from sentence vectors.

2.1 Edit Impact

Let (S, T) be a pair of ungrammatical and grammatical sentences, f be a function applying edits to an ungrammatical sentence, and $\mathcal{E}$ be a set of edits. Applying all edits in $\mathcal{E}$ to S obtains T, i.e., $T = f(S, \mathcal{E})$. We consider that an edit $e \in \mathcal{E}$ changing the meaning of a sentence drastically has a high impact. Then, we define an impact score I_e of e by the distance between a grammarical sentence T and another sentence $T_{-e} = f(S, \mathcal{E} \setminus e)$ that excludes an edit e from $\mathcal{E}$.

$$I_e = 1 - \frac{\mathrm{BERT}(T) \cdot \mathrm{BERT}(T_{-e})}{\|\mathrm{BERT}(T)\| \|\mathrm{BERT}(T_{-e})\|} \quad (1)$$

Here, $\mathrm{BERT}(T)$ presents a vector representation of the sentence T computed by the pre-trained BERT[2]. When we obtain a sentence $f(S, E)$ by applying a subset of edit operations $E \subseteq \mathcal{E}$, we define the overall impact score as the sum of the impact scores of all edits in E, i.e., $\sum_{e \in E} I_e$.

2.2 IMPARA Architecture

Considering the scores of both the CE and SE models for the input sentence S and the GEC output sentence O, we computes the overall score $\mathrm{score}(S, O) \in [0, 1]$. Denoting the correction score as $\mathrm{corr}(O)$, the similarity score as $\mathrm{sim}(S, O)$, and the threshold for the similarity score as θ, we define the overall score,

$$\mathrm{score}(S, O) = \begin{cases} \mathrm{corr}(O) & (\text{if } \mathrm{sim}(S, O) > \theta) \\ 0 & (\text{otherwise}) \end{cases}. \quad (2)$$

If the similarity score is less than or equal to θ, we regard that the output sentence is unrelated to the input sentence, and set the correction score to 0. In contrast, if the similarity score is greater than θ, we use the correction score as the overall score.

Correction score We compute the correction score as $\mathrm{corr}(O) = \sigma(R(O))$, where R presents the CE model and σ does the sigmoid function. We build R by fine-tuning a BERT model; more specifically, we model R as a linear transformation from the

[2]The mean of all token vectors in T at the final layer.

embeddings of the first token at the final layer to a scalar value. Hence, we describe the procedure for automatic construction of the supervision data (for training R) only from the parallel data of grammatical and ungrammatical sentences.

Let $\mathcal{C} = \{(S_i, T_i)\}_{i=1}^{n}$ be the parallel data of n instances of ungrammatical S_i and grammatical T_i sentences. For each instance $(S, T) \in \mathcal{C}$, we create pairs of pseudo edited sentences by applying different partial edits to S, and determine their order relations using the impact score (Eq. 1). In order to extract edit operations from (S, T), we find alignments using ERRANT (Bryant et al., 2017), and extract edits $E = \{e_1, \dots, e_{|E|}\}$ from S to T. We randomly create a subset $E^{'} \subseteq E$ with k elements, where $k \in \{1, 2, \dots, |E|\}$ is chosen from the discrete uniform distribution. Because comparing two subsets with large differences is difficult, we modify $E^{'}$ to create another subset $E^{''}$. We initialize $E^{''} = E^{'}$, and apply the following operation for each element $e \in E$ with the probability $\frac{1}{|E|}$.

$$E^{''} \leftarrow \begin{cases} E^{''} \cup \{e\} \text{ if } e \notin E^{'} \\ E^{''} \setminus \{e\} \text{ if } e \in E^{'} \end{cases} \tag{3}$$

We reject E' and E'' when this operation results in $E'' = E'$. In this way, we obtain psuedo edited sentences $f(S, E^{'}), f(S, E^{''})$ by applying $E^{'}$ and $E^{''}$ to the ungrammatical sentence S. We determine the order relation of the two sentence by using the impact score: we denote the edited sentence with a higher impact score as S_+ and the other as S_-. Generating at most c sentence pairs from a single pair of grammatical/ungrammatical sentences, we build the supervision data $\mathcal{T}$ for R.

We train R by minimizing the loss function L to learn the order of correction sentences.

$$L = \frac{1}{|\mathcal{T}|} \sum_{(S_-, S_+) \in \mathcal{T}} \sigma\left(R(S_-) - R(S_+)\right) \tag{4}$$

Here, we use the sigmoid function σ to avoid overweighting for some pairs in the supervision data[3].

Similarity score: To measure the semantic similarity between an input S and output O sentences, we calculate the cosine similarity $\mathrm{sim}(S, O)$ using the sentence vectors from a pre-trained BERT model.

[3]Preliminary experiments confirmed that the sigmoid function contirubted to improve the evaluation performance.

3 Experiments

3.1 Settings

We conduct two experiments for meta-evaluation of automatic evaluation metrics. The first evaluation assesses correlations between automatic and human evaluations on CoNLL-2014 dataset (Grundkiewicz et al., 2015), which is human-created ranking of the several GEC system outputs[4]. We compute Pearson's correlation (Pea) and Spearman's correlation (Spe) coefficients. We also measure accuracy (Acc) and Kendall's rank correlation coefficients (Ken) for sentence-level comparison. The CE model is trained on the parallel supervision data from CoNLL-2013 (Ng et al., 2013).

Second, we examine the ability of IMPARA to reflect domains and correction styles present in supervision data. We perform meta-evaluation with MAEGE (Choshen and Abend, 2018a)[5] on different combinations of supervision data for the CE model and meta-evaluation data. In these experiments, we use CWEB (Flachs et al., 2020) (website texts), FCE (Yannakoudakis et al., 2011) (essay), CoNLL-2014 (Ng et al., 2014) (minimal edits), and JFLEG (Napoles et al., 2017) (fluency edits).

We randomly sampled 90% of data for training, and used the remaining 10% for meta-evaluation. Pre-trained BERT[6] was used for the SE model, and fine-tuned for the CE model. We employ SOME (Yoshimura et al., 2020) and Scribendi Score (Islam and Magnani, 2021) as baselines. To verify the effectiveness of the construction method of the supervision data of IMPARA, we compare a CE model fine-tuned only on the sentence pairs of the original parallel corpus (only parallel). To train SOME, we used TMU dataset[7], with the same split as the holdout method in IMPARA and the hyperparameter settings of Yoshimura et al. (2020).

3.2 Results

Table 1 shows correlations between automatic and human evaluations[8]. IMPARA shows comparable correlations with SOME at sentence level, and outperforms SOME at corpus level. In the meta-evaluation of MAEGE (Table 2), IMPARA per-

[4]In this experiment, we used the Expected Wins.

[5]https://github.com/borgr/EoE

[6]https://github.com/huggingface/transformers

[7]https://huggingface.co/datasets/tmu_gfm_dataset

[8]As we could not reproduce Scribendi scores, we report the reported scores and ones computed by our implementation.

	Corpus		Sentence	
	Pea.	Spe.	Acc.	Ken.
Scribendi Score(ref.)	0.951	0.940	-	-
Scribendi Score(our impl.)	0.303	0.729	0.414	-0.170
SOME	0.956	0.923	**0.777**	**0.555**
IMPARA(only parallel)	0.936	0.929	0.742	0.485
IMPARA	**0.974**	**0.934**	0.748	0.496

Table 1: Correlation with manual evaluation on CoNLL-2014

	Corpus		Sentence		Chain
	Pea	Spe	Pea	Spe	Ken
Scribendi Score	0.884	0.981	0.374	0.421	**0.824**
SOME	**0.965**	**1.000**	0.394	0.439	0.563
IMPARA	0.951	0.990	**0.522**	**0.608**	0.692

Table 2: Meta-evaluation by MAEGE on CoNLL-2014

Eval	Train	Corpus		Sentence		Chain
		Pea	Spe	Pea	Spe	Ken
CoNLL 2013	CoNLL2013	0.932	**1.000**	**0.411**	**0.515**	**0.688**
	CWEB	0.961	**1.000**	0.380	0.468	0.574
	JFLEG	0.959	0.990	0.344	0.408	0.568
	FCE	**0.967**	**1.000**	0.404	0.490	0.567
CWEB	CoNLL2013	0.750	0.836	0.331	0.328	0.713
	CWEB	0.790	**0.963**	**0.472**	**0.432**	**0.780**
	JFLEG	0.757	0.818	0.353	0.354	0.775
	FCE	**0.805**	0.936	0.350	0.397	0.775
JFLEG	CoNLL2013	0.959	0.990	0.516	0.604	0.677
	CWEB	0.952	0.972	0.524	0.572	0.644
	JFLEG	0.937	**1.000**	**0.618**	**0.685**	**0.783**
	FCE	**0.961**	0.990	0.581	0.649	0.627
FCE	CoNLL2013	0.865	0.972	0.377	0.388	0.758
	CWEB	**0.882**	**0.990**	0.435	0.441	0.753
	JFLEG	0.852	0.972	0.390	0.429	0.739
	FCE	0.853	**0.990**	**0.541**	**0.616**	**0.848**

Table 3: Performance variation by combination of training and evaluation corpora

formed similarly to the baselines at corpus level, and outperformed the the baselines by up to 0.18 points in sentence-level and chain-level evaluations. These results indicate that IMPARA achieves the comparable or better evaluation performance than the existing reference-less methods, even with automatically generated supervision data.

Table 3 reports meta-evaluation using MAEGE on four evaluation corpora with diffirent training corpora. The results deminstrate that training and evaluating a CE model on the data of the same type improves the performance of automatic evaluation. Moreover, we compared the evaluation performance with existing methods using MAEGE (See table 4 in appendix). SOME and Scribendi Score sufferred from low performance on CWEB, FCE, and JFLEG. In contrast, IMPARA achieved the high performance in all evaluation corpora. This results suggest that IMPARA evaluates GEC outputs with characteristics of a dataset into consideration.

A further analysis indicates that correction impacts learned from parallel corpora focus more on corrections related to content words than those related to function words (see Section B in appendix).

4 Related Work

Major reference-based metrics include I-measure (Felice and Briscoe, 2015), M^2 (Dahlmeier and Ng, 2012), and ERRANT (Bryant et al., 2017) with precision, recall, and $F_{0.5}$ values. GLEU (Napoles et al., 2015) is based on BLEU metric (Papineni et al., 2002). These metrics require multiple references while IMPARA uses a single reference only.

Napoles et al. (2016) first introduced a reference-less method, which uses a detection tool of grammatical error and a language model. They showed that its performance was comparable to reference-based methods. Islam and Magnani (2021) proposed another method using GPT-2 (Radford et al., 2019). Although these methods require no supervision data for an evaluation model, we cannot adapt them to different domains or correction styles.

Asano et al. (2017) proposed a reference-less method, and outperformed reference-based methods by combining grammaticality, fluency, and meaning-preservation sub-metrics. This method uses regressor trained on GUG data (Heilman et al., 2014), language model and METEOR (Denkowski and Lavie, 2014) as sub-metrics. Yoshimura et al. (2020) showed that it was adequate to optimize an evaluation model for manual evaluation. However, they need costly data of human evaluation.

5 Conclusion

We proposed IMPARA, a method for constructing an automatic evaluation measure for GEC using a parallel corpus. The proposed method obtained a relative score for corrected sentences, utilizing impact scores of edits. We confirmed that IMPARA performed comparable or better than the existing methods in terms of correlations with human evaluation, and that it can perform automatic evaluation considering the characteristics of the used corpora.

Since IMPARA relies on parallel data, it needs parallel corpus corresponding to the domain or correction style of the evaluation data. Future work include construction of evaluation metrics without using parallel data, human evaluation data, multiple references, and treatment of mismatches of domain and/or correction styles.

References

Hiroki Asano, Tomoya Mizumoto, and Kentaro Inui. 2017. Reference-based metrics can be replaced with reference-less metrics in evaluating grammatical error correction systems. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 343–348, Taipei, Taiwan. Asian Federation of Natural Language Processing.

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.

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*, pages 5755–5762, New Orleans, Louisiana. Association for the Advancement of Artificial Intelligence.

Leshem Choshen and Omri Abend. 2018a. Automatic metric validation for grammatical error correction. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1372–1382, Melbourne, Australia. Association for Computational Linguistics.

Leshem Choshen and Omri Abend. 2018b. Inherent biases in reference-based evaluation for grammatical error correction. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 632–642, Melbourne, Australia. Association for Computational Linguistics.

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

Michael Denkowski and Alon Lavie. 2014. Meteor universal: Language specific translation evaluation for any target language. In *Proceedings of the Ninth Workshop on Statistical Machine Translation*, pages 376–380, Baltimore, Maryland, USA. Association for Computational Linguistics.

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*, pages 578–587, Denver, Colorado. Association for Computational Linguistics.

Simon Flachs, Ophélie Lacroix, Helen Yannakoudakis, Marek Rei, and Anders Søgaard. 2020. Grammatical error correction in low error density domains: A new benchmark and analyses. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8467–8478, Online. Association for Computational Linguistics.

Roman Grundkiewicz, Marcin Junczys-Dowmunt, and Edward Gillian. 2015. Human evaluation of grammatical error correction systems. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 461–470, Lisbon, Portugal. Association for Computational Linguistics.

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.

Kengo Hotate, Masahiro Kaneko, Satoru Katsumata, and Mamoru Komachi. 2019. Controlling grammatical error correction using word edit rate. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop*, pages 149–154, Florence, Italy. Association for Computational Linguistics.

Md Asadul Islam and Enrico Magnani. 2021. Is this the end of the gold standard? a straightforward reference-less grammatical error correction metric. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 3009–3015, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

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

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

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, Beijing, China. Association for Computational Linguistics.

Courtney Napoles, Keisuke Sakaguchi, and Joel Tetreault. 2016. There's no comparison: Reference-less evaluation metrics in grammatical error correction. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2109–2115, Austin, Texas. Association for Computational Linguistics.

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, Baltimore, Maryland. 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, Sofia, Bulgaria. Association for Computational Linguistics.

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

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.

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*, pages 180–189, Portland, Oregon, USA. Association for Computational Linguistics.

Ryoma Yoshimura, Masahiro Kaneko, Tomoyuki Kajiwara, and Mamoru Komachi. 2020. SOME: Reference-less sub-metrics optimized for manual evaluations of grammatical error correction. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 6516–6522, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Zheng Yuan and Ted Briscoe. 2016. Grammatical error correction using neural machine translation. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 380–386, San Diego, California. Association for Computational Linguistics.

Train and test data	Method	Corpus Pea	Corpus Spe	Sentence Pea	Sentence Spe	Chain Ken
CoNLL2013	Scribendi	0.938	0.984	0.331	0.355	**0.698**
	SOME	**0.961**	**1.000**	0.370	0.419	0.502
	IMPARA	0.932	**1.000**	**0.411**	**0.515**	0.688
CWEB	Scribendi	0.637	0.451	0.177	0.194	0.616
	SOME	0.767	0.663	0.055	0.155	0.678
	IMPARA	**0.790**	**0.963**	**0.472**	**0.432**	**0.780**
JFLEG	Scribendi	0.932	0.945	0.255	0.303	0.574
	SOME	**0.955**	0.990	0.523	0.531	0.639
	IMPARA	0.937	**1.000**	**0.618**	**0.685**	**0.783**
FCE	Scribendi	**0.869**	0.933	0.342	0.449	**0.897**
	SOME	0.843	0.972	0.165	0.254	0.663
	IMPARA	0.853	**0.990**	**0.541**	**0.616**	0.848

Table 4: Performance of IMPARA and existing methods using the same data for training and evaluation

Error type	Impact (10^{-2})	Frequency
NOUN	0.652	408
VERB:TENSE	0.649	480
VERB	0.580	557
NOUN:NUM	0.385	534
PUNCT	0.367	473
DET	0.364	1142
PREP	0.325	700

Table 5: Error types with frequency more than 400 (excluding OTHER) in CoNLL2014 and their assinged impact scores.

A Hyperparameters

To avoid the effect of the size of different corpora for fine-tuning the CE model during the comparisons, we adjusted the size of the training data to $|\mathcal{T}| = 4096$ regardless of the target corpus. We set the maximum number of edited sentence pairs generated from a pair of grammtical and ungramatical sentences to $c = 30$, the learning rate to 10^{-5}, and the batch size to 32. The number of epochs for fine-tuning varies from $1, 2, ..., 10$ to train the model. The threshold of similarity score θ is set to 0.9. We trained the models with four GPUs (RTX2080 Ti), performed a hyperparameter search on development set to select the best models.

B Impact on Different Error Types

We analyzed impact scores (defined in Section 2.1) assigned to different error types. For the sentence pairs in CoNLL-2014, we extracted edits and error types using ERRANT, and calculated the average impact score for each error type. Table 5 shows the averaged impact score for each error type that apperaed more than 400 times (excluding OTHER type).

As we expected, errors of content words such as NOUN (nouns) and VERB (verbs) were assigned with higher impact scores compared to those of functional words such as DET (determiner) and PREP (prepositions). In addition, we also observed that a lower impact score was calculated for corrections related to quantity. These results suggest that the impact score designed in this study is more concerned with changes in meaning caused by content words than with corrections related to grammatical roles caused by function words.