Training for Grammatical Error Correction Without Human-Annotated L2 Learners' Corpora

Mikio Oda
National Institute of Technology, Kurume College, Japan
oda@kurume-nct.ac.jp

Abstract

Grammatical error correction (GEC) is a challenging task for non-native second language (L2) learners and learning machines. Data-driven GEC learning requires as much human-annotated genuine training data as possible. However, it is difficult to produce larger-scale human-annotated data, and synthetically generated large-scale parallel training data is valuable for GEC systems. In this paper, we propose a method for rebuilding a corpus of synthetic parallel data using target sentences predicted by a GEC model to improve performance. Experimental results show that our proposed pre-training outperforms that on the original synthetic datasets. Moreover, it is also shown that our proposed training without human-annotated L2 learners' corpora is as practical as conventional full pipeline training with both synthetic datasets and L2 learners' corpora in terms of accuracy.

1 Introduction

Grammatical error correction (GEC) is one of the essential processes needed to produce sentences in a grammar-based language, and it is a challenging task for non-native second language (L2) learners and learning machines as well. Each language has its own grammar, however, data-driven language learning by a machine does not use the grammar, but corpora, more preferably, large-scale corpora. While classifiers that predict some token from candidates for a certain position in a sentence have been developed in the past (Li et al., 2019), sequence-to-sequence models have become more popular for GEC because the task is regarded as a sequence-to-sequence one and the models are flexible in editing sentences and covering various error types.

In sequence-to-sequence models, Felice et al. (2014) and Junczys-Dowmunt and Grundkiewicz (2014) treat the task as a statistical machine translation (SMT) problem and produce state-of-the-art performance on the CoNLL2014 shared task. Neural machine translation models (Sutskever et al., 2014), which consist of an encoder and a decoder, also have been investigated to improve their capabilities. In particular, the Transformer (Vaswani et al., 2017), which is an encoder-decoder model incorporating a self-attention mechanism, has become popular and various improved versions have been investigated. One of its alternative architectures is the Copy-Augmented Transformer, which has become popular for GEC (Hotate et al., 2020).

Another modification to the Transformer architecture is altering the encoder-decoder attention mechanism in the decoder to accept and make use of additional context. For example, Kaneko et al. (2019) use the BERT representation of the input sentence as additional context for GEC. GECToR (Omelianchuk et al., 2020) employs a BERT-like pre-trained encoder stacked with a linear layer with the softmax activation function, and treats the GEC task as a token labeling problem. Addressing training data for GEC models, Kiyono et al. (2019), Grundkiewicz et al. (2019) and Choe et al. (2019) employ synthetically generated pseudo data for pre-training of GEC systems prior to fine-tuning on human-annotated corpora for the Building Educational Applications (BEA) 2019 shared task (Bryant et al., 2019).

This paper addresses the effectiveness of synthetic parallel data, which is generally used as a consequence of the insufficiency of human-annotated L2 learners' corpora. We propose a method of substituting target sentences in synthetic parallel data with alternatives and rebuilding synthetic datasets to boost GEC training. Experiments demonstrate that pre-training on synthetic datasets rebuilt by the proposed method outperforms pre-training on the original synthetic datasets. Moreover, our synthetic datasets can be effectively employed not only to pre-train, but also to fine-tune GEC models, that is, training on synthetic data only

Proceedings of the 18th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2023), pages 455–465
July 13, 2023 ©2023 Association for Computational Linguistics

all through the pipeline. The GEC model's training without L2 learners' corpora is as practical as conventional training with both synthetic datasets and L2 learners' corpora in terms of accuracy.

2 Synthetic parallel training data

2.1 Generating synthetic training data

Supervised machine learning requires as much genuine training data as possible, and the same is true for GEC. Training data or corpora for GEC may be created with annotations by trained native speakers of the language or by grammarians. This fact makes it difficult for us to produce larger-scale genuine data, so researchers are compelled to use limited resources to train their learning models (Bryant et al., 2019). Therefore, synthetically generated large-scale parallel training data contributes to GEC systems along with the human-annotated data.

Synthetic parallel training data consist of erroneous sentences generated by corruption models from error-free sentences. In general, the corruption models can generate unlimited versions of erroneous sentences from a given error-free one, with the ability to vary the versions in the number of errors, error types, etc. Back-translation (Sennrich et al., 2016) provides monolingual training data with synthetic source sentences that are obtained from automatically translating the target sentence into the source language for NMT. Kiyono et al. (2019) apply back-translation to GEC and achieves state-of-the-art performance on the CoNLL2014 and BEA2019 test datasets.

PIE synthetic data (Awasthi et al., 2019) is often used in state-of-the-art GEC models proposed by Omelianchuk et al. (2020); Sorokin (2022), etc. Seq2Edits (Stahlberg and Kumar, 2020) is a sequence-to-sequence transducer which consists of a Transformer encoder and decoders, and can predict span-based edit operation probabilities for GEC. Stahlberg and Kumar (2021), furthermore, propose tagged corruption models using both Seq2Edits and a finite state transducer to match the observed error type distribution of the BEA2019 dev dataset, and generate synthetic data for pre-training GEC models.

2.2 Problems in synthetic training data

Given some noise to an error-free (grammatically correct) sentence, a system can generate a different version of the sentence which is generally regarded as a grammatically incorrect sentence. However, it does not always become an incorrect sentence. Table 1 shows some examples of inappropriate edits on the PIE-9M[1] and the C4-200M[2] synthetic datasets. The PIE model (Awasthi et al., 2019) and the tagged corruption model (Stahlberg and Kumar, 2021) each applies deletion to the source sentence, removing an adverb. In the PIE-9M synthetic dataset, the system removes the word *also* from the source sentence $\mathbf{y}^1$ to generate the erroneous sentence (Corrupted), and the edit to correct the sentence is *missing also* to recover from the error. However, the removed word is not necessarily required for the sentence $\mathbf{x}^1$ because it is an additive adverb, so the corrupted sentence $\mathbf{x}^1$ itself is an error-free sentence whose edit should be *no-operation*. The table also shows the same case in the C4-200M synthetic dataset. Note that *Source* is a target sentence to be outputted from a GEC model and *Corrupted* is a source sentence inputted to the model. The examples are cases where the original error-free sentences (Source) are inappropriate for the target sentences.

Large-scale synthetic parallel training datasets are often used to pre-train a GEC model prior to its fine-tuning on small-scale genuine datasets. The genuine datasets for the fine-tuning are annotated by trained native speakers of the language with respect to L2 learners' mistakes because the GEC model is expected to correct L2 learners' mistakes in text. Synthetic data for pre-training, therefore, should also match the data characteristics of L2 learners' grammatical mistakes as shown in human-annotated datasets to be employed in the final training. The corruption mechanism produces unexpected inappropriate edits on synthetic data that differ from human errors. Finally, synthetic data, itself, is one of the key resources for building better GEC systems.

3 Erroneous synthetic data rebuilt by GEC models

In this section, we further examine the problem described in the previous section and propose to rebuild conventional synthetic datasets, which are often employed by researchers, in order to create effective synthetic parallel training datasets for pre-training. A trained GEC model can be

[1] https://github.com/awasthiabhijeet/PIE/

[2] https://github.com/google-research-datasets/C4_200M-synthetic-dataset-for-grammatical-error-correction/

PIE 9M	
Source $\mathbf{y}^1$:	There have **also** been recent battles over access to multiple myeloma drug lenolidamide.
Corrupted:$\mathbf{x}^1$:	There have been recent battles to access to multiple myeloma drug lenolidamide.
Predicted $\tilde{\mathbf{y}}^1$:	There have been recent battles to access to multiple myeloma drug lenolidamide.
C4-200M	
Source $\mathbf{y}^2$:	We **just** have to live with black that are not truly black.
Corrupted $\mathbf{x}^2$:	We have to live in black that are not black.
Predicted $\tilde{\mathbf{y}}^2$:	We have to live in black that are not black.

Table 1: Examples of inappropriate edits of synthetic data for GEC. *Source* is an error-free sentence that is treated as a target sentence in a GEC training model. *Corrupted* is regarded as a grammatically incorrect sentence that is treated as a source sentence in the model. *Predicted* is a hypothetical target sentence generated by a GEC model. The bold words are not in the corrupted sentences; however, these words are not missing words that make the sentences ungrammatical.

represented by $g(\mathbf{x}^i)$, where $\mathbf{x}^i(= (x_1^i, \cdots, x_n^i))$ is the ith erroneous input sentence with tokens $x_j^i (1 \leq j \leq n)$, $g(\mathbf{x}^i)$ is the ith predicted output sentence: $\tilde{\mathbf{y}}^i = (\tilde{y}_1^i, \cdots, \tilde{y}_m^i)$. We train the model g with given datasets of incorrect and correct sentence pairs: $\mathcal{D} = \{(\mathbf{x}^i, \mathbf{y}^i) | i = 1, \cdots, N\}$, where the size of $\mathcal{D}$ is N, so as to decrease the difference (loss) of $\mathbf{y}^i$ between $\tilde{\mathbf{y}}^i$.

3.1 Process of generating synthetic data

Fig.1 shows a general process for generating synthetic parallel data consisting of an incorrect and correct sentence pair. The sentence $\mathbf{y}^i$ is an error-free sentence from a large-scale corpus such as Wikipedia, BookCorpus (Zhu et al., 2015) and the Colossal Clean Crawled Corpus (C4) (Raffel et al., 2020), and a corruption model produces some grammatical errors in the sentence $\mathbf{y}^i$ resulting in an erroneous sentence $\mathbf{x}^i$. The sentences $\mathbf{x}^i$ and $\mathbf{y}^i$ are the input sentence to a GEC model and the sentence that should be inferred by the model, respectively. The arrow from $\mathbf{y}^i$ to $\mathbf{x}^i$ is a noising process to add the errors, and the reverse dotted arrow is a de-noising process to restore the erroneous sentence to the correct form. In some cases, however, the target sentence of the noisy or erroneous sentence $\mathbf{x}^i$ should not be the unedited sentence $\mathbf{y}^i$, but another sentence $\hat{\mathbf{y}}^i$.

The noising and de-noising processes of the corruption models, therefore, often have irreversibility, and the hypothetically correct sentence $\hat{\mathbf{y}}^i$ does not always match the unedited error-free sentence $\mathbf{y}^i$. On the other hand, the process of generating a correct sentence $\hat{\mathbf{y}}^i$ from the erroneous sentence $\mathbf{x}^i$ by human annotators on genuine parallel data matches the correction process, and can create a dataset $\hat{\mathcal{D}} = \{(\mathbf{x}^i, \hat{\mathbf{y}}^i) | i = 1, \cdots, N\}$ which is significantly reliable as long as the annotators do not make mistakes. Even in human-annotated data, there can be plural candidates for the correct sentence $\hat{\mathbf{y}}^i$, but, the dataset $\hat{\mathcal{D}}$ is still reliable (Bryant et al., 2019).

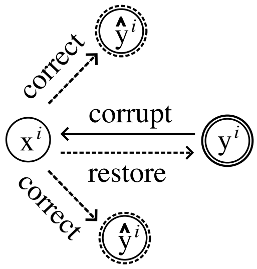

Figure 1: Process of generating a synthetic sentence pair $(\mathbf{x}^i, \mathbf{y}^i)$. The sentence $\mathbf{y}^i$ is an error-free sentence from a large-scale corpus. The sentence $\mathbf{x}^i$ is an erroneous sentence generated by a corruption model.

3.2 Proposed method for rebuilding synthetic data

We address synthetic data for GEC models and propose a modification where hypothetical target sentences are not original unedited sentences $\mathbf{y}^i$, but sentences predicted from corrupted ones by a conventional GEC model. In other words, we rebuild the synthetic data $\tilde{\mathcal{D}} = \{(\mathbf{x}^i, \tilde{\mathbf{y}}^i) | i = 1, \cdots, N\}$ from $\mathcal{D} = \{(\mathbf{x}^i, \mathbf{y}^i) | i = 1, \cdots, N\}$ which are usually used in pre-training of GEC models. This idea is similar to Rothe et al. (2021).

We employ a conventional GEC model $g(\mathbf{x}^i)$ to generate hypothetical target sentences $\tilde{\mathbf{y}}^i$. One would expect that the predicted sentences $\tilde{\mathbf{y}}^i$ from corrupted sentences $\mathbf{x}^i$ by a GEC model would match the corrected sentences $\hat{\mathbf{y}}^i$: $\tilde{\mathbf{y}}^i \simeq \hat{\mathbf{y}}^i$ for $\mathbf{x}^i$, and build an appropriate synthetic dataset : $\tilde{\mathcal{D}} \simeq \hat{\mathcal{D}}$. The conventional GEC model we employ in this paper is GECToR (Omelianchuk et al., 2020), where the number of labels is $5,004$. GECToR has achieved state-of-the-art results on GEC, however, the version of the model we employ achieves $\mathrm{F}_{0.5}$ scores of 64.0 and 71.8 on the CoNLL 2014 and BEA 2019 test datasets, respectively. As the GEC systems, of course, are still under development by researchers, we have to compromise on the quality of synthetic data rebuilt by our proposed method. Table 1 also shows examples of hypothetical target sentences $\tilde{\mathbf{y}}^i$, which contain grammatical errors, generated by the GEC model.

3.3 Synthetic data rebuilt by the GEC model

To predict $\tilde{\mathbf{y}}^i$ from $\mathbf{x}^i$ we employ a newer version of the trained GECToR model[3] which has a RoBERTa encoder based on the results of Omelianchuk et al. (2020) and the inference hyperparameters, *confidence bias* and *minimum probability* threshold, are set to 0.2 and 0.5, respectively. As synthetic data to be examined, we use the above-mentioned PIE-9M and C4-200M in the experiments; the former is widely used for pre-training GEC models and the latter is generated by attempting to match the error type frequency distribution to the development dataset. Note that the C4-200M dataset is downsized to 9M sentences to match the size of the PIE-9M in the experiments.

Table 2 shows the fundamental statistics of the synthetic datasets rebuilt by the proposed method, compared to the original ones. The average numbers of tokens per sentence in the rebuilt datasets $\tilde{\mathcal{D}}$s are not significantly different from those of the original datasets $\mathcal{D}$s. To compare statistical relationships between sentences $\mathbf{x}^i$ and $\mathbf{y}^i$, we generate m2 formatted information using the ERRor ANnotation Toolkit (ERRANT)[4](Bryant et al., 2017) and calculate the average number of edits per sentence. Applying the proposed method to the PIE-9M and C4-200M train datasets, the procedure reduces the average number of edits (corruptions) per sentence, resulting in about 0.8 and 2.8 fewer than the original datasets, respectively. We also indicate the dataset $\check{\mathcal{D}}$, which has a comparable average number of edits with the dataset $\tilde{\mathcal{D}}$. The erroneous sentences $\check{\mathbf{x}}^i$ are generated from the corrupted sentences $\mathbf{x}^i$ in the PIE-9M dataset by recovering edits partly to adjust its average of edits to that of the dataset $\tilde{\mathcal{D}}$. The dataset $\check{\mathcal{D}}$ is used in the experiments in the next section to prove that the effectiveness of our method does not depend on the number of edits per sentence empirically.

Stahlberg and Kumar (2021) have tried to match their synthetic data characteristics to L2 learners' error characteristics with respect to the frequency of occurrence of the error types for the reason that the trained model is mainly expected to correct L2 learners' sentences. We further examine whether our method can regulate the frequency of occurrence with respect to grammatical error types in the synthetic datasets to match the L2 learners'. Fig.2 shows the frequency distribution of occurrence with respect to grammatical error types in our rebuilt synthetic datasets $\tilde{\mathcal{D}}$s, comparing the original synthetic datasets $\mathcal{D}$s, PIE-9M and C4-200M, and L2 learners' corpus, the Cambridge English Write & Improve (W&I+LOCNESS) v2.1[5](Bryant et al., 2019; Granger, 1998). The proposed method changes the frequency of error occurrence, and we expect that the frequency distribution of $\mathcal{D}$ could approach that of the L2 learners' corpus by the proposed method. Note that the L2 learners' corpus for comparison is employed in stage III training of GEC models, which is the final fine-tuning stage in the experiments, and the corpus for the final stage of training is of utmost importance.

To investigate the similarity between two frequency distributions, we calculate Kullback-Leibler (KL) divergence, which is a measure of how different two probability distributions are from each other, defined as

$$D_{\mathrm{KL}}(P||Q) = \sum_{x \in \chi} P(x) \log \left(\frac{P(x)}{Q(x)} \right), \qquad (1)$$

where P and Q are discrete probability distributions and χ is the sample space. We consider the frequency distributions as the probability distributions, and the sample space χ is 24 error types defined by ERRANT. Table 2 also shows the average level of information, i.e., entropy. The entropy measures uncertainty of the types of grammatical errors that will occur in a sentence.

Comparing each entropy value of the proposed synthetic datasets $\tilde{\mathcal{D}}$s with that of their original ones $\mathcal{D}$s, the proposed method approaches the entropy of the PIE-9M synthetic data and that of the W&I LOCNESS dataset $\mathcal{D}_{WI}$, while there is no

[3] https://github.com/grammarly/gector/

[4] https://github.com/chrisjbryant/errant/

[5] https://www.cl.cam.ac.uk/research/nl/bea2019st/

Synthetic (#sentences)	Dataset	$\mathbf{x}^i/\mathbf{y}^i/D$	#tokens	#edits	Entropy [bit]	$D_{\mathrm{KL}}(\mathcal{D}_{\mathrm{WI}}\|\cdot)$	$D_{\mathrm{KL}}(\mathcal{D}_{\mathrm{Co}}\|\cdot)$
PIE-9M (8.42M)		$\mathbf{x}^i$	25.1	—	—	—	—
		$\check{\mathbf{x}}^i$	25.2	—	—	—	—
		$\mathbf{y}^i$	25.4	—	—	—	—
		$\hat{\mathbf{y}}^i$	25.1	—	—	—	—
	Original	$\mathcal{D}(\mathbf{x}^i, \mathbf{y}^i)$	—	2.45	3.79	0.216	**0.198**
	Proposed	$\tilde{\mathcal{D}}(\mathbf{x}^i, \tilde{\mathbf{y}}^i)$	—	1.60	3.87	**0.186**	0.216
	Random	$\check{\mathcal{D}}(\check{\mathbf{x}}^i, \mathbf{y}^i)$	—	1.62	3.79	0.198	0.216
C4-200M (8.42M)		$\mathbf{x}^i$	25.7	—	—	—	—
		$\mathbf{y}^i$	25.7	—	—	—	—
		$\hat{\mathbf{y}}^i$	25.8	—	—	—	—
	Original	$\mathcal{D}(\mathbf{x}^i, \mathbf{y}^i)$	—	4.04	3.80	**0.093**	**0.177**
	Proposed	$\tilde{\mathcal{D}}(\mathbf{x}^i, \tilde{\mathbf{y}}^i)$	—	1.26	3.80	0.196	0.369
W&I+LOC		$\mathcal{D}_{\mathrm{WI}}(\mathbf{x}^i, \mathbf{y}^i)$	—	—	3.88	—	0.128
CoNLL2014		$\mathcal{D}_{\mathrm{Co}}(\mathbf{x}^i, \mathbf{y}^i)$	—	—	3.86	0.143	—

Table 2: The statistical metrics of the sentences and the datasets, where $\mathbf{x}^i$ are corrupted sentences from the corresponding error-free sentences $\mathbf{y}^i$ in the original corpus. The proposed method creates hypothetical correct sentences $\tilde{\mathbf{y}}^i$ of the dataset $\tilde{\mathcal{D}}$. The comparative partially recovered sentences $\check{\mathbf{x}}^i$ are created to match the average number of edits per sentence to the proposed $\tilde{\mathcal{D}}$. W&I+LOC is its train dataset and CoNLL2014 is its test dataset.

significant difference from the C4-200M dataset. In the PIE-9M synthetic dataset, the proposed method also approaches the frequency distribution of the types of grammatical errors to that of $\mathcal{D}_{WI}$. Regarding the C4-200M dataset, on the other hand, the proposed method moves the frequency distribution away from that of $\mathcal{D}_{WI}$, however, the two datasets rebuilt by the proposed method, $\tilde{\mathcal{D}}$s, have almost the same value of KL divergence from $\mathcal{D}_{WI}$. The table also refers to the values of KL divergence from the CoNLL2014 dataset for evaluating the GEC models. Note that the CoNLL2014 dataset is small-sized and consists of 1,312 sentences.

4 Experiments

To empirically investigate the effectiveness of the proposed method and the capabilities of a GEC model trained on synthetic data rebuilt by the method, we train the GEC model choosing the hyperparameters described below. The GEC model is fundamentally trained through the three stage pipeline adopted in Choe et al. (2019), Omelianchuk et al. (2020), Stahlberg and Kumar (2021), etc.: stage I is a pre-training stage on a synthetic dataset, stage II is a training stage on a human-annotated dataset and stage III is a fine-tuning stage on a smaller human-annotated dataset more consistent with the target domain of GEC.

4.1 Training model and datasets

In the experiments, we employ RoBERTa (Liu et al., 2019)(roberta-base[6]) and train the model on the datasets indicated below. Hyperparameters in the training stage are set to the same values as on the website[7] (Omelianchuk et al., 2020), and choosing a set of labels to be predicted by the model is done in the same manner as described there. We also employ three different PIE-9M and three different C-200M datasets.

Stage I (Pre-training) Either the PIE-9M or the C-200M is used in stage I as a conventional method. Each dataset consists of 9M sentence pairs, which we randomly split into two sets: 95% train and 5% dev datasets. The data splitting creates 8.42M sentence-pair synthetic parallel datasets $\mathcal{D}$s. We apply the proposed method to the above datasets $\mathcal{D}$s to create the proposed synthetic parallel datasets $\tilde{\mathcal{D}}$s. We also create the dataset $\check{D}$ which has a similar average number of edits per sentence by recovering some edits randomly and partially to adjust to the statistics of the proposed datasets. The statistical information for all the synthetic parallel datasets is shown in Table 2. Note that all text in the C-200M dataset is tokenized using spaCy and the en_core_web_sm model[8].

[6] https://huggingface.co/models/

[7] https://github.com/grammarly/gector/

[8] https://spacy.io/

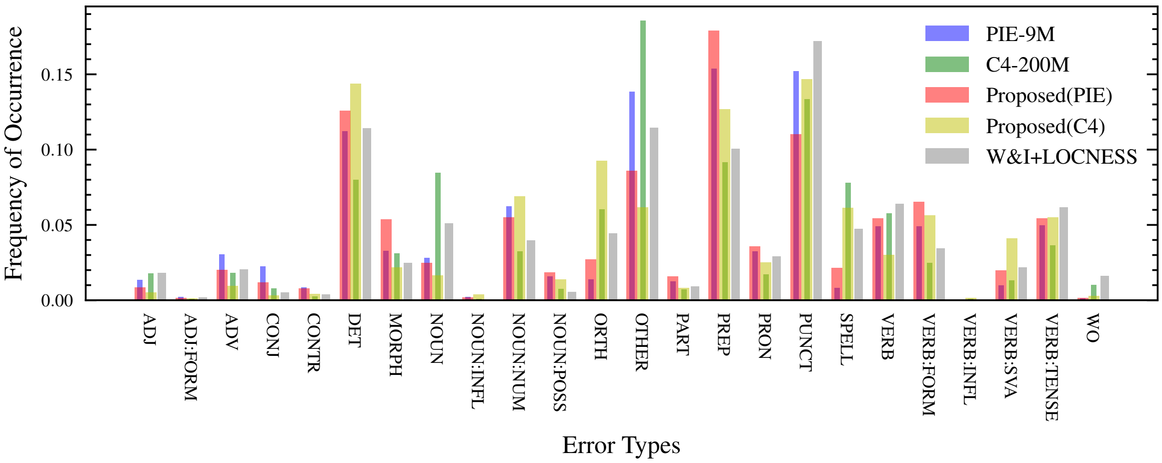

Figure 2: The frequency distribution of occurrence with respect to grammatical error types defined by ERRANT. ERRANT analyzes grammatical errors, which are categorized into 24 types, in sentences $\mathbf{x}^i$ by comparing those to target sentences $\mathbf{y}^i$ or $\tilde{\mathbf{y}}^i$. The statistics of the synthetic datasets rebuilt by the proposed method are compared with the original synthetic datasets and L2 learners' corpus.

Stage II (Training) We employ L2 learners' human-annotated corpora used in the BEA2019 shared task. The corpora consist of W&I+LOCNESS v2.1, the First Certificate in English (FCE) v.2.1 (Yannakoudakis et al., 2011), the National University of Singapore Corpus of Learner English (NUCLE) (Dahlmeier et al., 2013) and the Lang-8 Corpus of Learner English (Lang-8) (Mizumoto et al., 2011; Tajiri et al., 2012) shown in Table 3. We split the corpora into 98% train and 2% dev datasets because they are small-sized and train data of a larger size is preferable. Table 3 shows the characteristics of each corpus and the overall corpus for stages II and III.

Stage III (Fine-tuning) We choose W&I+LOCNESS, one of the corpora in stage II, as an L2 learners' corpus consistent with the target domain of GEC. This selection is based on Choe et al. (2019) for the restricted track and Omelianchuk et al. (2020). In addition to the L2 learners' corpus, the synthetic dataset rebuilt by the proposed method is downsized to 34K sentence pairs for fine-tuning of the models pre-trained on the same synthetic data. The sentence pairs of the downsized synthetic dataset are chosen randomly from the 9M sentence pairs.

4.2 Results

We trained the GEC models on either of the original PIE-9M, C4-200M or our rebuilt synthetic datasets in stage I followed by training in a combination of stages II and III. Both stages II and III use the L2 learners' corpora or our rebuilt 34K synthetic datasets described in Sec. 4.1. To evaluate the performance of the trained models, we let each model correct grammatical errors in the sentences of the CoNLL2014 and BEA2019 test datasets. Note that we set the *confidence bias* and the *minimum probability* threshold to zeros for inference after stages I and II as on the website. We evaluated the performance of the models for the CoNLL2014 and BEA2019 test datasets using M^{2}scorer[9] and by submitting the corrected sentences to the server

Dataset	Stage	#sents	#tokens	#edits
W&I+L A	II, III	10,490	17.5	2.69
W&I+L B	II, III	13,030	18.3	1.83
W&I+L C	II, III	10,781	19.2	0.926
FEC	II	28,345	16.0	1.52
NUCLE	II	56,957	20.3	0.758
Lang-8	II	1.04M	11.4	1.20
total(train)	II	1.13M	12.2	1.24
total(train)	III	33,614	18.3	1.84

Table 3: L2 learners' corpora employed in training stages II and/or III. The number of sentence pairs, the average number of tokens and edits per sentence are indicated for each corpus. Each corpus is split into train and dev datasets, and the overall train data for stages II and III is also shown. Note that *sentence* means a token sequence to be inputted to the model, and each sentence in the W&I+LOCNESS is assigned to a CEFR level, A, B or C.

[9] https://github.com/nusnlp/m2scorer/

referred to by the BEA2019 shared task website[10], respectively.

Table 4 shows comparisons of GEC performance with metrics, precision (P), recall (R) and $F_{0.5}$ scores for the test datasets, indicating train datasets each model used in stages I, II and III. The results for the PIE-9M synthetic dataset are summarized as follows. The baselines are the underlined results of the model trained on the conventional datasets, that is, *Original+BEA2019* through stages I, II and III, resulting in $F_{0.5} = 62.9$ and $F_{0.5} = 70.5$ for the CoNLL2014 and BEA2019 test datasets, respectively. While the pre-trained *Original* performs $F_{0.5} = 51.2$ and $F_{0.5} = 51.1$, the pre-trained *Proposed* performs $F_{0.5} = 61.2$ and $F_{0.5} = 66.7$, respectively. For the partial pipeline training of stages I and III, the *Original+BEA2019* performs $F_{0.5} = 62.4$ and $F_{0.5} = 70.3$, and the *Proposed+BEA2019* performs $F_{0.5} = 62.8$ and $F_{0.5} = 70.1$, respectively. *Proposed+PIE-34K*, which was pre-trained and fune-tuned only on the rebuilt PIE-9M and PIE-34K synthetic datasets, performs $F_{0.5} = 62.9$ and $F_{0.5} = 71.5$, respectively. *Proposed+C4-34K* was pre-trained and fine-tuned only on the synthetic datasets as well, however, the training employed two different synthetic datasets, PIE and C4. For the full pipeline training of stages I, II and III, the *Original+BEA2019* performs $F_{0.5} = 62.9$ and $F_{0.5} = 70.5$, and the *Proposed* performs $F_{0.5} = 63.6$ and $F_{0.5} = 70.6$, respectively. The results regarding the C4-200M synthetic dataset are also shown in the same manner in the figure.

5 Discussion and related work

This paper addresses the *quality* of synthetic parallel data due to the insufficiency of human-annotated L2 learners' corpora and the effectiveness of training only on synthetic data. Note that the *quality* does not address grammatical correctness, but the validity of source-target sentence pairs for training and how well the data fits the characteristics of L2 learners' mistakes. The overall results indicate that our method is more effective for the PIE-9M dataset than the C4-200M dataset, and it implies that the C4-200M dataset is of better quality.

Here, we discuss the experiments on the PIE-9M dataset, which more likely needs the technique. The stage-I training by the proposed method outperforms the conventional training by 10.0 and 15.6 with regard to $F_{0.5}$ for the CoNLL2014 and BEA2019 test datasets, respectively. It results in only 1.7 and 3.8 less than the baselines, which were trained through the full pipeline, stages I, II, and III.

Furthermore, the stage-II training reduces the performance of the pre-trained models on the proposed method's synthetic dataset. This suggests that the proposed method's synthetic datasets could be of higher quality than the overall L2 learners' corpora while each synthetic dataset itself could be inferior to the L2 learners' corpora.

Unfortunately, the baseline of the training replaced with the rebuilt synthetic dataset does not improve its performance. Our synthetic datasets can be employed all through the pipeline of training, that is, training without L2 learners' corpora. The results show that GEC model training without L2 learners' corpora is as practical as conventional training with both L2 learners' corpora and synthetic datasets in terms of accuracy. Note that the version of the model employed to rebuild synthetic data in the experiments achieves the scores of 64.0 and 71.8 on $F_{0.5}$ for the CoNLL 2014 and BEA 2019 test datasets, respectively.

To summarize the achievements, the proposed method :

1. outperforms pre-training on the original synthetic datasets.
2. provides notably good training performance without human-annotated L2 learners' corpora.

Trained GEC models can be used not only for predicting correct sentences but also for generating better synthetic data, and systems incorporating the proposed method are not limited to the synthetic data and model used in this paper.

Addressing training data for GEC models, Grundkiewicz and Junczys-Dowmunt (2014) introduce the WikEd Error Corpus generated from Wikipedia revision histories, corpus content and format. The corpus consists of more than 12 million sentences with a total of 14 million edits of various types. Kiyono et al. (2019), Grundkiewicz et al. (2019) and Choe et al. (2019) employ synthetically generated pseudo data for pre-training of GEC systems prior to fine-tuning on human-annotated corpora for the BEA2019 shared task(Bryant et al., 2019).

[10] https://www.cl.cam.ac.uk/research/nl/bea2019st/

Synthetic	Training Datasets	Stage			CoNLL2014 test			BEA2019 test		
		I	II	III	P	R	$F_{0.5}$	P	R	$F_{0.5}$
PIE-9M	Original	$\mathcal{S}$			60.4	31.8	51.2	54.3	41.5	51.1
	PartiallyRecovered	$\mathcal{S}$			58.7	32.3	50.4	51.2	42.9	49.3
	Proposed	$\mathcal{S}$			66.5	46.3	**61.2**	68.3	60.9	**66.7**
	Original+BEA2019	$\mathcal{S}$	$\mathcal{A}$		64.2	45.3	**59.2**	61.3	58.5	**60.7**
	PartiallyRecovered+BEA2019	$\mathcal{S}$	$\mathcal{A}$		63.7	45.5	59.0	60.0	59.3	59.8
	Proposed+BEA2019	$\mathcal{S}$	$\mathcal{A}$		64.0	45.1	59.1	60.0	58.8	59.8
	Original+BEA2019	$\mathcal{S}$		$\mathcal{A}$	72.4	40.2	62.4	76.4	53.4	70.3
	PartiallyRecovered+BEA2019	$\mathcal{S}$		$\mathcal{A}$	72.8	39.1	62.1	76.4	52.3	70.0
	Proposed+BEA2019	$\mathcal{S}$		$\mathcal{A}$	70.7	43.5	62.8	74.3	57.0	70.1
	Proposed+PIE-34K	$\mathcal{S}$		$\mathcal{S}$	73.9	39.5	**62.9**	78.1	53.5	71.5
	Proposed+C4-34K	$\mathcal{S}$		$\mathcal{S}$	75.1	37.5	62.5	79.7	52.2	**72.1**
	Original+BEA2019	$\mathcal{S}$	$\mathcal{A}$	$\mathcal{A}$	69.1	46.8	62.9	75.0	56.6	70.5
	PartiallyRecovered+BEA2019	$\mathcal{S}$	$\mathcal{A}$	$\mathcal{A}$	73.3	42.1	**63.9**	75.0	56.5	70.4
	Proposed+BEA2019	$\mathcal{S}$	$\mathcal{A}$	$\mathcal{A}$	73.4	41.5	63.6	75.8	55.3	**70.6**
C4-200M	Original	$\mathcal{S}$			64.2	39.1	56.9	62.9	50.4	59.9
	Proposed	$\mathcal{S}$			66.3	47.9	**61.6**	68.1	62.0	**66.8**
	Original+BEA2019	$\mathcal{S}$	$\mathcal{A}$		65.6	46.3	**60.6**	61.2	60.5	**61.0**
	Proposed+BEA2019	$\mathcal{S}$	$\mathcal{A}$		63.7	45.9	59.1	59.6	59.5	59.5
	Original+BEA2019	$\mathcal{S}$		$\mathcal{A}$	72.5	42.1	63.3	78.1	56.3	**72.5**
	Proposed+BEA2019	$\mathcal{S}$		$\mathcal{A}$	70.9	44.7	63.4	73.1	58.8	69.7
	Proposed+C4-34K	$\mathcal{S}$		$\mathcal{S}$	75.3	40.0	**64.0**	77.9	54.6	71.8
	Proposed+PIE-34K	$\mathcal{S}$		$\mathcal{S}$	74.8	39.9	63.6	78.2	54.3	71.8
	Original+BEA2019	$\mathcal{S}$	$\mathcal{A}$	$\mathcal{A}$	72.9	43.1	**64.0**	75.8	58.3	**71.5**
	Proposed+BEA2019	$\mathcal{S}$	$\mathcal{A}$	$\mathcal{A}$	73.4	41.4	63.6	75.1	56.3	70.4

Table 4: Comparison of GEC performance after pre-training (stage I) on either the original synthetic datasets or the datasets rebuilt by the proposed method. The pre-trained models were further trained on either the L2-learners' corpora or the rebuilt synthetic datasets in stages II and/or III. $\mathcal{S}$ and $\mathcal{A}$ mean $\mathcal{S}$ynthetic and $\mathcal{A}$nnotated train datasets, respectively.

Mita et al. (2020) focus on human annotators' errors in official datasets when they rewrite incorrect sentences to remove grammatical mistakes and denoise the target sentences of the official datasets using some trained GEC models with a perplexity criterion. Rothe et al. (2021) also apply the similar technique to the LANG-8 corpus, which is a large corpus of texts written by L2 learners with user-annotated corrections, and correct human errors by the GEC models.

Our proposed method is effective not only for correcting human annotators' errors, but also for adjusting source-target disparity to match the domain. Stahlberg and Kumar (2021) build a large synthetic pre-training dataset with error tag frequency distributions matching Seq2Edits (Stahlberg and Kumar, 2020). Parnow et al. (2021) trained a generator to generate increasingly realistic errors (in the form of token-based edit labels) and a discriminator to differentiate between artificially-generated edits and human-annotated edits. Stahlberg and Kumar (2021) propose tagged corruption models using both the Seq2Edits and a finite state transducer to match the observed error type distribution of the BEA2019 dev dataset, and generate synthetic data for pre-training GEC models. Yasunaga et al. (2021) apply BIFI algorithm (Yasunaga and Liang, 2021) and LM-Critic to synthetic data to generate better datasets for GEC. LM-Critic chooses the most likely grammatical sentence from multiple sentence candidates based on the sentence occurrence probabilities generated by a language model.

6 Conclusion

In this paper, we have addressed the effectiveness of synthetic parallel data and have proposed a method for rebuilding a corpus of synthetic parallel data using target sentences predicted by a GEC model. While the original target sentences in synthetic parallel data are guaranteed to be error-free, the target sentences predicted by a GEC model contain grammatical errors because the GEC model has been developed through research and is not perfect in its performance. However, pre-training on our proposed synthetic data outperforms that on the original synthetic data, and our pre-trained GEC model showed performance only slightly lower than the conventional fine-tuned GEC model. In addition, our proposed method can provide notably good training performance without human-annotated L2 learners' corpora.

The proposed method's target sentences by an imperfect GEC model work better than the original error-free target sentences although the former may contain grammatical errors. The reason why this paradoxical result happens needs to be determined. In future work, we plan to investigate further reconfiguration and modification of synthetic parallel data, and fine-tune training using such data to improve the performance of GEC. Investigation of the source-target relationships on training data mentioned above should also be carried out to clarify the effectiveness of the proposed method.

Acknowledgements

We gratefully thank Martin Chodorow at CUNY Hunter College for his valuable suggestions and feedback. Furthermore, we would like to thank the reviewers for their insightful comments. This work was supported by JSPS KAKENHI (Grant Numbers JP18K00904 and JP21K00806).

References

Abhijeet Awasthi, Sunita Sarawagi, Rasna Goyal, Sabyasachi Ghosh, and Vihari Piratla. 2019. Parallel iterative edit models for local sequence transduction. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4260–4270, Hong Kong, China. Association for Computational Linguistics.

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, Florence, Italy. Association for Computational Linguistics.

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.

Yo Joong Choe, Jiyeon Ham, Kyubyong Park, and Yeoil Yoon. 2019. A neural grammatical error correction system built on better pre-training and sequential transfer learning. In *Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 213–227, Florence, Italy. 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, Atlanta, Georgia. Association for Computational Linguistics.

Mariano Felice, Zheng Yuan, Øistein E. Andersen, Helen Yannakoudakis, and Ekaterina Kochmar. 2014. Grammatical error correction using hybrid systems and type filtering. In *Proceedings of the Eighteenth Conference on Computational Natural Language Learning: Shared Task*, pages 15–24, Baltimore, Maryland. Association for Computational Linguistics.

Sylviane Granger. 1998. *The computer learner corpus: A versatile new source of data for SLA research*. 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 *Advances in Natural Language Processing*, pages 478–490, Cham. Springer International Publishing.

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.

Kengo Hotate, Masahiro Kaneko, and Mamoru Komachi. 2020. Generating diverse corrections with local beam search for grammatical error correction. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 2132–2137, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Marcin Junczys-Dowmunt and Roman Grundkiewicz. 2014. The AMU system in the CoNLL-2014 shared task: Grammatical error correction by data-intensive and feature-rich statistical machine translation. In *Proceedings of the Eighteenth Conference on Computational Natural Language Learning: Shared Task*, pages 25–33, Baltimore, Maryland. Association for Computational Linguistics.

Masahiro Kaneko, Kengo Hotate, Satoru Katsumata, and Mamoru Komachi. 2019. TMU transformer system using BERT for re-ranking at BEA 2019 grammatical error correction on restricted track. In *Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 207–212, Florence, Italy. Association for Computational Linguistics.

Shun Kiyono, Jun Suzuki, Masato Mita, Tomoya Mizumoto, and Kentaro Inui. 2019. An empirical study of incorporating pseudo data into grammatical error correction. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1236–1242, Hong Kong, China. Association for Computational Linguistics.

Ruobing Li, Chuan Wang, Yefei Zha, Yonghong Yu, Shiman Guo, Qiang Wang, Yang Liu, and Hui Lin. 2019. The LAIX systems in the BEA-2019 GEC shared task. In *Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 159–167, Florence, Italy. Association for Computational Linguistics.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A robustly optimized BERT pretraining approach. *arXiv preprint arXiv:1907.11692*.

Masato Mita, Shun Kiyono, Masahiro Kaneko, Jun Suzuki, and Kentaro Inui. 2020. A self-refinement strategy for noise reduction in grammatical error correction. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 267–280, Online. 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.

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.

Kevin Parnow, Zuchao Li, and Hai Zhao. 2021. Grammatical error correction as GAN-like sequence labeling. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 3284–3290, Online. Association for Computational Linguistics.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.

Sascha Rothe, Jonathan Mallinson, Eric Malmi, Sebastian Krause, and Aliaksei Severyn. 2021. A simple recipe for multilingual grammatical error correction. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 702–707, Online. Association for Computational Linguistics.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Improving neural machine translation models with monolingual data. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 86–96, Berlin, Germany. Association for Computational Linguistics.

Alexey Sorokin. 2022. Improved grammatical error correction by ranking elementary edits. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 11416–11429, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Felix Stahlberg and Shankar Kumar. 2020. Seq2Edits: Sequence transduction using span-level edit operations. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5147–5159, Online. Association for Computational Linguistics.

Felix Stahlberg and Shankar Kumar. 2021. Synthetic data generation for grammatical error correction with tagged corruption models. In *Proceedings of the 16th Workshop on Innovative Use of NLP for Building Educational Applications*, pages 37–47, Online. Association for Computational Linguistics.

Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. 2014. Sequence to sequence learning with neural networks. In *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'14, page 3104–3112, Cambridge, MA, USA. MIT Press.

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 (Volume 2: Short Papers)*, pages 198–202, Jeju Island, Korea. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, ¥L ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

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.

Michihiro Yasunaga, Jure Leskovec, and Percy Liang. 2021. LM-critic: Language models for unsupervised grammatical error correction. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7752–7763, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Michihiro Yasunaga and Percy Liang. 2021. Break-it-fix-it: Unsupervised learning for program repair. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 11941–11952. PMLR.

Y. Zhu, R. Kiros, R. Zemel, R. Salakhutdinov, R. Urtasun, A. Torralba, and S. Fidler. 2015. Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. In *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 19–27, Los Alamitos, CA, USA. IEEE Computer Society.