TransGEC: Improving Grammatical Error Correction with Translationese

Anonymous ACL submission

Abstract

Data augmentation is an effective way to improve model performance of grammatical error correction (GEC). This paper identifies a critical side-effect of GEC data augmentation, which is due to the style discrepancy between the data used in GEC tasks (i.e., texts produced by non-native speakers) and data augmentation (i.e., native texts). To alleviate this, we propose to use an alternative data source, translationese (i.e., human-translated texts), as input for GEC data augmentation, which 1) is easier to obtain and usually has better quality than non-native texts, and 2) has a more similar style to non-native texts. Experimental results on the CoNLL14 and BEA19 English, NLPCC18 Chinese, Falko-MERLIN German, and RULEC-GEC Russian GEC benchmarks show that our approach consistently improves correction accuracy over strong baselines. Further analyses reveal that our approach is helpful for overcoming mainstream correction difficulties such as the corrections of frequent words, missing words, and substitution errors. Source code and scripts will be released.

1 Introduction

Grammatical error correction (GEC) is a task to automatically correct an ungrammatical sentence into a corrected version. Training GEC models highly relies on labeled data (i.e., ungrammatical sentences to their grammatical ones), but such resources are scarce and expensive to construct. Data augmentation, which exploits a large amount of unlabeled data for performance improvement, is a popular research line of GEC (Rozovskaya and Roth, 2010; Felice et al., 2014; Rei et al., 2017; Kasewa et al., 2018; Xie et al., 2018). However, there is a style mismatch between the data used in GEC tasks and data augmentation. For most GEC tasks (Ng et al., 2014), their training and testing instances are produced by non-native speakers, whereas the data used for augmentation are mainly native language resources (Kiyono et al., 2019; Zhao et al., 2019; Grundkiewicz et al., 2019; Kaneko et al., 2020a). Rabinovich et al. (2016) have shown a large difference between non-native and native texts, which means that style mismatch might be a side-effect limiting the further enhancement of GEC data augmentation. A more ideal way is directly using non-native texts as input for data augmentation. However, such resources are very few, and their qualities are hard to be guaranteed.

In this paper, we propose the TransGEC method which uses human-translated texts (aka translationese) as input for augmentation. Improving GEC with translationese has the following advantages: 1) **easy-to-obtain**, the training corpus of machine translation tasks consists of abundant translationese, and its identification has been well studied (Riley et al., 2020); 2) **similar style**, non-native texts and translationese are closer to each other than native texts (Rabinovich et al., 2016); and 3) **high quality**, most translationese is produced by bilingual experts, whose quality can be better guaranteed than the majority of non-native texts.

Preliminary experiments on the comparison of different kinds of texts confirm our assumption that translationese indeed has a similar style to GEC data. This enables us to further explore translationese for GEC in two steps: 1) obtaining translationese, we propose to fine-tune BERT-based classifiers to identify translationese from the parallel corpora (e.g., WMT corpus) of machine translation tasks; and 2) improving GEC with translationese, we propose to add artificial noise to the identified translationese, and treat the noisy/corrected version as the input/output for training GEC models.

Experimental results on the widely-used CoNLL14 and BEA19 English, NLPCC18 Chinese, Falko-MERLIN German, and RULEC-GEC Russian GEC benchmarks show that our approach improves the model performance over strong (m)T5-large pre-trained model (Raffel et al., 2019; Xue

et al., 2020), LRGEC baselines (Náplava and Straka, 2019), and existing data augmentation methods (Zhao et al., 2019). Further analyses show that our TransGEC method improves the correction accuracy of major difficulties (e.g., correction of frequent words, missing words, and substitution errors), but still has room for improvement in minor issues (e.g., correction of rare words, word order, and deletion errors).

Our main contributions are summarized as:

- We empirically show that translationese has a similar style to the original GEC data in different languages (i.e., English and Chinese).
- We introduce how to simply obtain translationese and propose a novel method, TransGEC, to improve GEC with translationese.
- We confirm the effectiveness of exploiting translationese as input for GEC data augmentation with and without pre-trained models.
- We reveal the linguistic properties enhanced and diminished after exploiting translationese, providing some clues for future studies.

2 Related Work

Grammatical Error Correction (GEC) can be viewed as a kind of sequence-to-sequence learning task (Sutskever et al., 2014; Yuan and Briscoe, 2016; Ji et al., 2017; Chollampatt and Ng, 2018; Junczys-Dowmunt et al., 2018). Since labeled training data is scarce and hard to collect, various data augmentation methods are proposed to enhance the performance of GEC models. Kasewa et al. (2018), Xie et al. (2018) and Kiyono et al. (2019) use the back-translation method (Sennrich et al., 2016) to produce the noisy data for GEC. Zhao et al. (2019) and Lichtarge et al. (2019) use certain noise rules to inject wrong information into correct sentences. Kiyono et al. (2019) give an empirical study of incorporating synthetic data for GEC. Stahlberg and Kumar (2021) exploit the error tagged corruption model to generate synthetic data.

Another research line uses pre-trained language models to improve the model performance of GEC. Kaneko et al. (2020b) extract external knowledge from language models for GEC training, Rothe et al. (2021) further treat the language models as a part of the network for GEC training. All the above work has a potential limitation: while the training and test data of GEC tasks are produced by non-native speakers, the data used for augmentation or pre-training are mainly native texts. This style discrepancy is a threat to GEC data augmentation.

Madnani et al. (2012) and Zhou et al. (2020) propose to use machine-translated text for GEC data augmentation, but their intuition is not using the text with a similar style but producing noisy text through machine translation. Our approach focuses on the style mismatch problem by introducing translationese (human-translated texts) as input for data augmentation, providing a reasonable explanation for their model improvements.

Translationese refers to the presence of unusual properties of human-translated texts and thus becomes an alternate name for such texts. A reason might be that translators are affected by the style of the source language and ignore the rules of the target language during translation (Gellerstam, 1986). Translationese tends to show less lexical diversity compared to native texts (Stubbs, 1996). Britt et al. (2015) point out that there are many common idioms unconsciously used in native texts. Baker et al. (1993) and Toury (1995) report that translationese has some unique characteristics, e.g., simplification, explicitation and normalization. Rabinovich et al. (2016) provide a systematic study and find that the non-native texts and translationese are closer to each other than to native texts.

A research line discusses the effect of translationese in machine translation tasks since translationese widely exists in parallel corpora. Graham et al. (2020) reveal the side-effect of using translationese in machine translation evaluation and recommend only native texts for machine translation evaluation. Riley et al. (2020) demonstrate that translationese hinders the model from generating more adequate and fluent translations.

Another line focuses on identifying translationese from parallel sentences to control the training of downstream tasks. Kurokawa et al. (2009) propose a support vector machine-based classifier to identify translationese while Riley et al. (2020) use a convolution neural network-based classifier. Wang et al. (2021) train a classifier based on native and translationese data differ significantly at the text content to distinguish between them.

To the best of our knowledge, the discussion and application of translationese has not yet been introduced to GEC tasks. This paper takes the first step into using translationese for improving GEC.

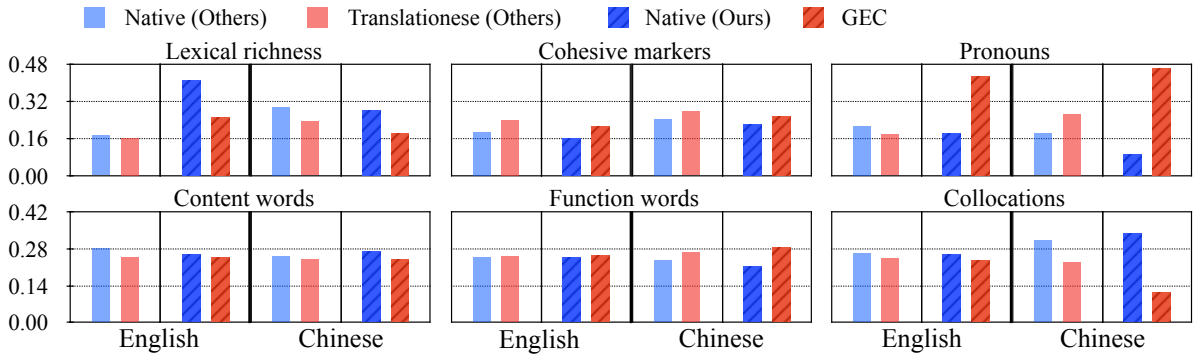

Figure 1: Four kinds of texts in English and Chinese languages. Native (Others) and Translationese (Others) represent our reproduced results based on the released English data by Rabinovich et al. (2016) and the Chinese data by McEnery and Xiao (2004) and Xiao et al. (2008). Native (Ours) refers to the results based on our collected native text (i.e., the WMT News Crawl data for English and the People's Daily data for Chinese), and GEC refers to the results of the original GEC data (i.e., CoNLL14 English and NLPCC18 Chinese benchmarks). The vertical axis represents the normalized statistical results for each linguistic property, where a higher value indicates a greater proportion of linguistic properties. **The style of translationese is similar to that of original GEC data.**

3 Why Translationese?

We first explain why GEC models need other kinds of alternatives as input for data augmentation, and then give preliminary experiments and results to show that translationese can be a decent alternative.

Motivation The performance of GEC systems highly depends on the quality and quantity of annotated training data (i.e., ungrammatical sentences and their grammatical version). Due to the high cost of collecting such data, the research of data augmentation techniques (i.e., utilizing unlabeled data) for GEC has become a popular topic.

By looking at the most widely-used GEC benchmark – CoNLL14 (Ng et al., 2014) and BEA19 (Bryant et al., 2019) shared tasks, the training corpora includes NUS Corpus of Learner English (NUCLE) (Dahlmeier et al., 2013), Lang-8 Corpus (Tajiri et al., 2012), FCE v2.1 (Yannakoudakis et al., 2011) and W&I (Yannakoudakis et al., 2018), all of which are produced by non-native language learners. However, existing methods directly use native texts as input for data augmentation for GEC tasks. For example, Kiyono et al. (2019) and Kaneko et al. (2020a) use Wikipedia data, while Zhao et al. (2019) and Grundkiewicz et al. (2019) utilize One Billion Word Benchmark (Chelba et al., 2013) data.

Previous studies have validated that there exists a style gap between native and non-native texts (Rabinovich et al., 2016). We argue that such gap brings a side-effect to model performance, limiting the further improvements of GEC data augmentation. Utilizing non-native texts might be a better choice, however, there exist few non-native text resources and it is not easy to collect the text from scratch and guarantee their quality. This motivates us to find some other alternatives, which are **easy-to-obtain**, **high-quality**, and with **a closer style** to the non-native text of GEC tasks.

Preliminary Experiments Rabinovich et al. (2016) have shown that non-native texts and translationese are closer to each other than each of them to native texts. Motivated by them, in this experiment, we explore the similarities between GEC data and translationese on the English and Chinese GEC tasks. We compare our collected native texts and GEC data on the properties of lexical richness, cohesive markers, collocations, pronouns, content words, and function words. To make a fair comparison, we directly use the same data provided by Rabinovich et al. (2016) and Su and Li (2016) to reproduce the results of native texts and translationese. The settings are shown in Appendix A.1.

As shown in Figure 1, the trend of our collected native texts and GEC data is consistent with that of the native texts and translationese provided by existing work. For example, both the translationese and GEC data are of lower lexical richness and contain more cohesive markers and function words than the native texts. One outlier is the result of English pronouns, and the reason is the overuse of personal pronouns such as 'I' and 'you' in the GEC data. However, by looking at the result of Chinese pronouns, it still has the same trend. The above results confirm our assumption that translationese and GEC data have a similar style than native texts.

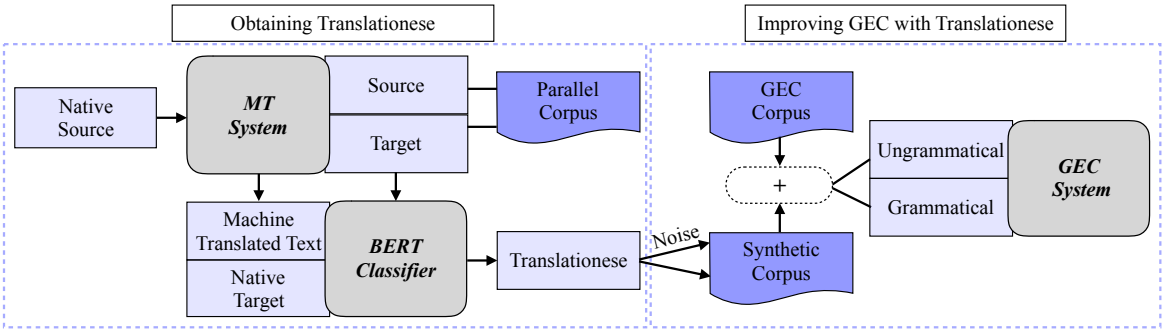

Figure 2: The overall framework of TransGEC. The left half is to obtain translationese from the target side of the parallel corpus, and the right half is to use the obtained translationese as input for GEC data augmentation. Specifically, the native source monolingual text is translated to machine translated text through a trained machine translation system. The translationese is identified via the BERT classifier, which is fine-tuned with the same amount of machine translated text and native target monolingual text. The obtained translationese is injected with specific noise to produce a synthetic GEC corpus which is merged with the original GEC corpus to train a GEC system.

4 TransGEC

The above observations enable us to further improve GEC with translationese. Figure 2 shows the overall framework of TransGEC, which contains two parts: obtaining translationese and improving GEC with translationese.

Obtaining Translationese Existing parallel corpora of machine translation (MT) tasks (Bojar et al., 2017) have a huge amount of translationese on both sides. However, most parallel corpora do not annotate whether an instance is native or translated. Therefore, previous studies (Kurokawa et al., 2009; Riley et al., 2020) have had to train a classifier to identify and obtain translationese from parallel corpora. In this paper, to obtain translationese from existing parallel training corpora of MT, we propose to fine-tune BERT-based classifiers using a small number of machine translated texts (Devlin et al., 2019), which can alleviate the limitation of Riley et al. (2020) relying on a large amount of machine translated texts to train a convolutional neural network-based classifier from scratch.

Specifically, given a parallel corpus $\mathcal{D}_{\text{mt}} = \{(x^n, y^n)\}_{n=1}^{N}$, we first need to train a machine translation model $f_{x \mapsto y}$ that translates a source sentence x to a target sentence y:

$$f_{x \mapsto y} : \arg\max_{\theta} \left\{ \sum_{n=1}^{N} \log P\left(y^n | x^n; \theta\right) \right\} \quad (1)$$

Then, the machine translated texts $\mathcal{Y}_{\text{mt}}$ can be obtained by translating the native source sentences:

$$\mathcal{Y}_{\text{mt}} = \{ f_{x \mapsto y}(x) \mid x \in \mathcal{X}_{\text{native}} \} \quad (2)$$

where $\mathcal{X}_{\text{native}}$ denotes native source texts, which can be easily collected (e.g., WMT News Crawl).

Given the generated $\mathcal{Y}_{\text{mt}}$ and collected $\mathcal{Y}_{\text{native}}$, we fine-tune the BERT-based pre-trained language model as a classifier to distinguish whether a sentence is native or not. After that, we use the fine-tuned BERT-based classifier to label the target side of the parallel corpus $\mathcal{D}_{\text{mt}}$, and identify the sentences which have lower classification probabilities to be native texts as translationese $\mathcal{Y}_{\text{trans}}$.

Improving GEC with Translationese This part exploits the obtained translationese $\mathcal{Y}_{\text{trans}}$ as input for GEC data augmentation. Motivated by Zhao et al. (2019), artificial noise is added to $\mathcal{Y}_{\text{trans}}$ and the synthetic GEC corpus $\mathcal{D}_{\text{syn}}$ can be viewed as:

$$\mathcal{D}_{\text{syn}} = \{ (\delta(y), y) \mid y \in \mathcal{Y}_{\text{trans}} \} \quad (3)$$

where $\delta(\cdot)$ denotes the noise operator with the following four types of noise: 1) deletion, randomly delete a token in the sentence; 2) insertion, randomly add a token into a sentence; 3) replacement, randomly select a token from the vocabulary to replace a token in the sentence; 4) word order, shuffle the words in the sentence by a Gaussian distribution bias and then subsequently reorder the sentence.

After that, we can train a GEC model with the original corpus $\mathcal{D}_{\text{gec}}$ and synthetic corpus $\mathcal{D}_{\text{syn}}$:

$$\arg\max_{\theta} \left\{ \sum\nolimits_{(s,t) \in \mathcal{D}_{\text{gec}} \cup \mathcal{D}_{\text{syn}}} \log P\left(t | s\right) \right\} \quad (4)$$

where s denotes a noisy (ungrammatical) sentence and t denotes its corresponding corrected (grammatical) version. The model parameters θ can be randomly intialized or intialized from large-scale pre-trained language models.

5 Experiments

5.1 Obtaining Translationese

Setup We conduct experiments on English, German, Russian and Chinese. We treat WMT17 News Crawl data in English, German and Russian as their native texts, and use Chinese News[1] as Chinese native texts. We deduplicate and filter sentences whose lengths are longer than 70 tokens. The pre-trained Chinese⇒English translation model (Wu et al., 2019) is used to generate English machine translated texts from native Chinese News. To obtain German, Russian and Chinese machine translated texts, we translate the native English texts using the pre-trained English⇒German (Ott et al., 2018) and English⇒Russian (Ng et al., 2019), and our own English⇒Chinese translation models (37.7 BLEU (Papineni et al., 2002) on newstest17).

We use 1M native texts and 1M machine translated texts to fine-tune the BERT-based translationese classifiers (Devlin et al., 2019) for each language. The settings of fine-tuning BERT-based classifiers are listed in Appendix A.2. We use the classifiers to identify translationese and native texts from the target side of the UN Chinese⇔English and UN English⇒Russian (Ziemski et al., 2016) corpora, and WMT16 English⇒German corpora.

Results The confidence threshold of identifying translationese (native texts) is set to >0.9 (<0.1). We evaluate the fine-tuned BERT-based classifiers by F_1 score on WMT test sets, which consist of native texts and translationese in equal number (Zhang and Toral, 2019). Compared to Riley et al. (2020) score of $0.85F_1$ on the English⇒German newstest15, our classifier achieved $0.91F_1$ on the same test set. For English, Chinese and Russian, our classifiers score $0.94F_1$, $0.80F_1$, and $0.85F_1$ on the Chinese⇒English newstest17, English⇒Chinese newstest17, and English⇒Russian newstest17, respectively.

Finally, 6.9M English and 5.8M Chinese translationese are selected from the UN corpus. Due to the small amount of the training data for German and Russian GEC tasks, we sample 50K Russian and 120K German translationese from the UN Russian and WMT16 German, respectively. We present classified examples in Appendix A.3.

[1] https://github.com/brightmart/nlp_chinese_corpus

Language	Corpus	Train	Dev	Test
EN	BEA19	0.56M	-	-
EN	W&I	-	3,396	3,447
EN	LOCNESS*	-	988	1,030
EN	cLang8-en	2.4M	-	-
EN	CoNLL13	-	1,379	-
EN	CoNLL14	-	-	1,312
ZH	NLPCC18	1.09M	5,000	2,000
DE	Falko-MERLIN	12.9K	2.503	2,337
RU	RULEC-GEC	4,980	2,500	5,000

Table 1: Statistics of the used data sets. Data marked with * is native while the others are non-native data.

5.2 Improving GEC with Translationese

Data We use the BEA19 workshop official dataset (Bryant et al., 2019) for our preliminary experiments. The training data of BEA19 are non-native texts, including FCE v2.1 (Yannakoudakis et al., 2011), Lang-8 Corpus of learner English (Mizumoto et al., 2011; Tajiri et al., 2012), NUCLE (Dahlmeier et al., 2013) and W&I (Yannakoudakis et al., 2018). While the development and test sets of BEA19 consist of W&I and LOCNESS (Granger, 1998), W&I consists of 3 different levels of non-native texts and LOCNESS is native text. Specifically, we use W&I dev and LOCNESS dev as the validation sets when testing the performance on the W&I test set and LOCNESS test set, respectively.

For the main English experiments, we use the distilled cLang-8 corpus as the training data, which is a clean version of Lang-8 data (Rothe et al., 2021). The CoNLL13 (Ng et al., 2013) and the widely used *official-2014.combined.m2* version of CoNLL14 (Ng et al., 2014) are used for validation and test sets, respectively. For Chinese, we use the official training and test data of NLPCC18 (Zhao et al., 2018), which are also produced by second language learners. We follow Zhao and Wang (2020) to randomly select a subset from the training data as the development set. For German and Russian, we use the same 10M systhetical dataset as Náplava and Straka (2019) for pretraining and then follow them to finetune on the Falko-MERLIN (Boyd et al., 2014) German dataset and RULEC-GEC (Rozovskaya and Roth, 2019) Russian dataset, these datasets are also the learner corpora. Table 1 presents the statistics of the data we used.

For generating synthetic data, we corrupt the translationese with four certain rules: deletion, insertion, replacement, and word order. For the first three rules, we conduct several groups of exper-

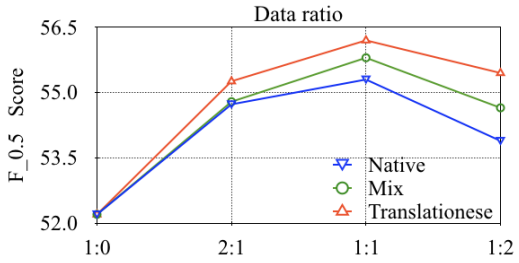

Figure 3: Results of the different types of synthetic data combined with original cLang-8 GEC data with different combination ratios on the CoNLL14 test set.

MODEL	METHOD	W&I	LOCNESS	ALL
TRANSF.	BASE	53.7	33.7	51.7
	+NATIVE	54.3	**35.9**	52.5
	+MIX	55.3	35.3	53.1
	+TRANS.	**56.0**	34.5	**53.4**

Table 2: $F_{0.5}$ scores on the BEA19 English benchmark. BASE uses the original BEA19 training data. ALL is the full BEA19 test set. +NATIVE can be seen as combining the native texts with base GEC data, +TRANS. (Trans-GEC method) means translationese, and +MIX refers half of the native texts and half of the translationese. **Bold** values indicate the best results.

iments to explore the best setting of corruption probabilities and find that setting them to 0.05, 0.1, 0.2 works well (see Appendix A.4). For word order, we shuffle the words by adding a Gaussian bias to their positions and then reorder the words with a standard deviation of 0.5.

Models and Training For preliminary English experiments, the GEC models are based on the Transformer architecture and implemented using the open-source toolkit fairseq (Ott et al., 2019). We follow the default TRANSFORMER-BASE settings to initialize our model with a shared embedding. The other settings are listed in Appendix A.5. The main experiments for English and Chinese, which are based on the T5 (Raffel et al., 2019) and mT5 (Xue et al., 2020) models of their large variants. We follow Rothe et al. (2021) to fine-tune the pre-trained models on English cLang-8 GEC data. In addition, we fine-tune the pre-trained models on Chinese GEC data. The details of the fine-tuning settings T5 and mT5 are listed in Appendix A.6.

For German and Russian, we follow Náplava and Straka (2019) to use TRANSFORMER-BIG architecture and implement using tensor2tensor (Vaswani et al., 2018) toolkit. As regards the pretraining and finetuning procedure and the prameters, we also follow their repository.[2]

The M2 scorer (Dahlmeier and Ng, 2012) is used for evaluating our models on CoNLL14 English, Falko-MERLIN German, RULEC-GEC Russian, and NLPCC18 Chinese GEC tasks. The ERRANT scorer (Bryant et al., 2019) is used for evaluating on BEA19 English task. We run experiments with three different random seeds and report the averaged scores. To test the significance of the results, we adopt the T-test method in the SciPy toolkit.[3]

[2] https://github.com/ufal/low-resource-gec-wnut2019

[3] https://scipy.org

Augmentation ratio Before conducting the experiments, we first investigate the effect of the proportion of synthetic data on the model performance model. As shown in Figure 3, there are three types of data: Native, Tanslationese, Mix (mixture of native texts and translationese). We combine them with the original cLang-8 GEC data using different ratios settings (i.e., 1:0, 2:1, 1:1, 1:2). When the ratio is set to 1:1, all the data groups consistently achieve the best performance than the other settings. Therefore, the experiments in the subsequent sections directly use the augmentation ratio of 1:1.

Preliminary Results Table 2 presents the $F_{0.5}$ results of the BEA19 English GEC task. The Transformer model trained with translationese (i.e., +TRANS.) achieves the best result on the BEA19 non-native W&I and ALL test set. While testing on the BEA19 native LOCNESS test set, the model trained with native texts (i.e., +NATIVE) achieves the best $F_{0.5}$ scores. It confirms our assumption that using the texts with a similar style for GEC data augmentation is beneficial to GEC tasks.

Main Results Table 3 presents the results of the CoNLL14 English, NLPCC18 Chinese, Falko-MERLIN German, and RULEC-GEC Russian GEC tasks. It can be seen that all the three types of synthetic data outperform the baselines, confirming the effectiveness of GEC data augmentation. Moreover, the models trained with translationese (i.e., **+TRANS.**) achieve the best precision and $F_{0.5}$ scores over **BASELINE** and **+NATIVE** models on the English, Chinese, German and Russian GEC tasks, respectively. Notably, the reported results for German and Russian are based on the strong baselines LRGEC (Náplava and Straka, 2019) because the (M)T5 LARGE baselines are slightly weaker(see Appendix A.7). For comparability, we randomly select half of the native texts and half of the trans-

Model (Method)	EN (CoNLL14)			ZH (NPLCC18)			DE (Falko-MERL.)			RU (RULEC-GEC)		
	Pre.	Rec.	$F_{0.5}$	Pre.	Rec.	$F_{0.5}$	Pre.	Rec.	$F_{0.5}$	Pre.	Rec.	$F_{0.5}$
MaskGEC	-	-	-	44.4	22.2	37.0	-	-	-	-	-	-
TagGEC	72.8	49.5	66.6	-	-	-	-	-	-	-	-	-
LRGEC	-	-	63.4	-	-	-	78.2	59.9	73.7	63.3	27.5	50.2
(m)T5 large	-	-	66.0	-	-	-	-	-	70.1	-	-	27.6
(m)T5 xxl	-	-	68.8	-	-	-	-	-	74.8	-	-	43.5
Our Baseline	71.8	51.4	66.5	41.5	25.8	37.0	77.6	61.0	73.6	64.9	26.3	50.2
+Native	73.2	51.4	67.5	43.6	24.6	37.8	78.2	62.1	74.3	65.3	26.3	50.4
+Mix	73.8	51.2	67.8	43.1	**26.5**	38.3	78.6	62.1	74.6	65.1	26.8	50.6
+Trans.	**74.7**	**51.6**	**68.6**†	**45.2**	24.5	**38.7**†	**78.8**	**62.2**	**74.8**††	**65.4**	**26.8**	**50.8**†

Table 3: Results on CoNLL14 English, NLPCC18 Chinese, Falko-MERLIN German, and RULEC-GEC Russian GEC tasks. **Baseline** refers to the GEC training data. Native texts and translationese are identified from the same domain. **+Native** can be seen as the proposed method by Zhao et al. (2019), who use native texts for augmentation. **+Trans.** refers to the synthetic data generated from translationese. **+Mix.** means the synthetic data is made up of half of native texts and half of translationese. (m)T5 large/xxl results indicate the models fine-tuned on cLang8 GEC data, which was reported by Rothe et al. (2021). Notably, our results for German and Russian are based on the strong baseline LRGEC (Náplava and Straka, 2019). Statistically significant improvements over **+Native** method are reported using P_value, $^{\dagger}p < 0.01$, $^{\dagger\dagger}p < 0.05$. **Bold** values indicate the best scores.

lationese (i.e., **+Mix**) to train the GEC models. The results show that the $F_{0.5}$ scores are higher than **+Native** but lower than **+Trans.** models. Overall, our proposed TransGEC approach, i.e., **+Trans.**, outperforms other popular methods. Although the recall score of the **+Trans.** models is not the highest, the evaluation of GEC tasks usually pays more attention to the precision and $F_{0.5}$ score, since neglecting a correction is not as bad as proposing a wrong correction (Ng et al., 2014). Appendix A.8 shows examples produced by Native and Translationese English GEC models. We also report the results on the BEA19 test set in Appendix A.9. The results present the same trend. The reason is that the translationese keeps the style consistent with the original GEC training data, making the GEC models learning knowledge much easier.

Compared to Existing Methods The MaskGEC (Zhao and Wang, 2020) model dynamically inserts noise to the source sentences for GEC. It is a strong baseline for Chinese NLPCC18 benchmark. TagGEC (Stahlberg and Kumar, 2021) uses an error tagged corruption model to produce synthetic data for the GEC task. LRGEC (Náplava and Straka, 2019) focuses on GEC in low resource scenarios and utilizes synthetic parallel data to improve them. (m)T5 large/xxl (Rothe et al., 2021) is fine-tuned on (m)T5 large/xxl pre-trained models with the same cLang-8 data used in experiments (i.e., Base). From Table 3, our proposed method (i.e., **+Trans.**) based on the strong (m)T5 large and LRGEC baselines consistently improves correction accuracy for English, Chinese, German and Russian GEC benchmarks.

6 Analysis

In this section, we analyze our results from two perspectives: error types and linguistic properties.

Error Types We investigate the performance of different error types for English and Chinese GEC tasks. We use the ERRANT toolkit (Bryant et al., 2017) for English. For Chinese, we use the adapted ERRANT released by Hinson et al. (2020). As shown in Table 4, the GEC system augmented with translationese performs well in correcting all types of errors. For Chinese, the GEC system augmented with translationese is good at correcting missing words, and substitution errors. The performance gap between Chinese and English might be caused by their different sentence structures. Our approach is more effective to improve the correction accuracy of the major difficulties, i.e., missing words (17.9%/38.0%), and substitution errors (64.3%/54.0%) on the English/Chinese GEC benchmarks. However, there is still some room for improvement in minor issues (e.g., correction of word order and deletion errors).

Linguistic Properties We investigate two linguistic properties in terms of word frequency and position. The detailed settings are presented in Appendix A.10. As shown in Figure 4, **+Native** and

Language	Error Type	Ratio	Native			Mix			Translationese		
			Pre.	Rec.	$F_{0.5}$	Pre.	Rec.	$F_{0.5}$	Pre.	Rec.	$F_{0.5}$
English	Word Order	0.8%	34.1	40.5	35.2	34.4	40.0	35.4	35.9	40.1	**36.7**
	Deletion	17.0%	45.8	27.9	40.6	47.2	28.1	41.6	49.4	26.8	**42.3**
	Missing	17.9%	40.0	26.4	36.3	40.5	26.3	36.5	40.2	27.8	**36.9**
	Substitution	64.3%	45.9	22.3	37.9	45.4	22.4	37.7	46.0	22.7	**38.2**
Chinese	Word Order	2.9%	38.9	37.8	**38.7**	37.7	37.7	37.7	37.4	39.0	37.7
	Deletion	5.1%	5.9	27.9	**7.0**	5.8	28.4	6.9	5.8	27.6	6.9
	Missing	38.0%	27.3	19.2	25.2	27.5	18.9	25.2	28.2	19.9	**26.0**
	Substitution	54.0%	31.9	13.8	25.3	32.5	14.2	25.8	33.2	15.1	**26.8**

Table 4: Performance by error types when using different kinds of texts for augmentation. We give the ratio of each type. **Bold** values indicate the best $F_{0.5}$ score in each row. The model augmented with translationese has a better ability in correcting missing words and substitution errors.

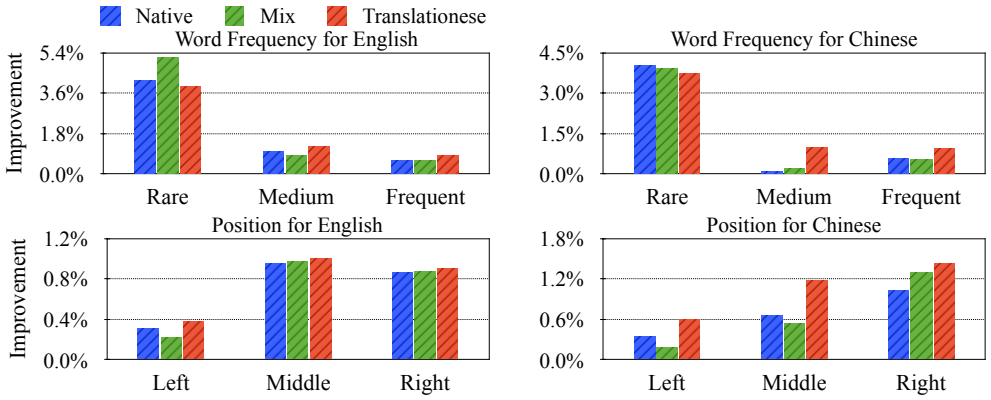

Figure 4: Improvements of exploiting different types of texts for augmentation in terms of word frequency and position on the English and Chinese GEC tasks. Overall, the translationese method (i.e., TransGEC) can bring more benefits to GEC in terms of linguistic properties. We discuss the outlier of correcting rare words in the text part.

+MIX methods are better than **+TRANS.** method to correct rare words, but fail to correct the words with higher frequency. The reason might be that the lexical diversity of native texts is higher than translationese. Furthermore, we count the proportion of frequent/medium/rare tokens for the training data, which are 90.3%/6.1%/3.6 for English and 91.7%/5.3%/3.0 for Chinese. It means our method can better alleviate the major issue of GEC tasks.

In terms of position, the improvement of the left position is lower than those of the middle and right in the English/Chinese GEC task. It might be that English and Chinese are the right-branching languages that usually describe the main subject first and provide the key information at the tail of the sentence to explain the subject (Payne, 2006). It may be also that the middle and right parts of the sentences benefit from more previous context. The result of **+TRANS** GEC system is consistently superior to **+NAITVE** GEC system. This confirms that using the augmentation data with a similar style to GEC data is beneficial to GEC models.

7 Conclusion

This paper introduces a TransGEC method that uses translationese as input for data augmentation of GEC. Preliminary experiments on native texts, translationese, and GEC data confirm that the translationese and GEC data share a similar style compared to native texts. Based on the evidence, we propose a simple and effective method to mine translationese from parallel corpora by classifiers and construct a synthetic GEC corpus by adding artificial noise to the translationese. Experimental results on the CoNLL14 and BEA19 English, NLPCC18 Chinese, Falko-MERLIN German, and RULEC-GEC Russian benchmarks show that the models augmented with translationese can outperform strong baselines. Further analyses show that our approach performs well in solving major difficulties (e.g., correction of frequent words, missing words, and substitution errors), but still has some room for improvement in minor issues (e.g., correction of rare words, word order, and deletion errors).

References

Mona Baker, Gill Francis, and Elena Tognini-Bonelli. 1993. Corpus linguistics and translation studies: Implications and applications. In *Text and Technology: In Honour of John Sinclair*, Netherlands. John Benjamins Publishing Company.

Ondřej Bojar, Rajen Chatterjee, Christian Federmann, Yvette Graham, Barry Haddow, Shujian Huang, Matthias Huck, Philipp Koehn, Qun Liu, Varvara Logacheva, Christof Monz, Matteo Negri, Matt Post, Raphael Rubino, Lucia Specia, and Marco Turchi. 2017. Findings of the 2017 conference on machine translation (WMT17). In *Proceedings of the Second Conference on Machine Translation*, pages 169–214, Copenhagen, Denmark. Association for Computational Linguistics.

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

Erman Britt, Annika Denke, Fant Lars, and Forsberg Lundell Fanny. 2015. Nativelike expression in the speech of long-residency l2 users: A study of multiword structures in l2 english, french and spanish. *International Journal of Applied Linguistics*, 25:160–182.

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.

Ciprian Chelba, Tomas Mikolov, Mike Schuster, Qi Ge, Thorsten Brants, Phillipp Koehn, and Tony Robinson. 2013. One billion word benchmark for measuring progress in statistical language modeling. *arXiv preprint arXiv:1312.3005*.

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, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*, pages 5755–5762. AAAI Press.

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.

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.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. 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.

Martin Gellerstam. 1986. Translationese in swedish novels translated from english. *Translation studies in Scandinavia*, 1:88–95.

Yvette Graham, Barry Haddow, and Philipp Koehn. 2020. Statistical power and translationese in machine translation evaluation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 72–81, Online. Association for Computational Linguistics.

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

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.

Charles Hinson, Hen-Hsen Huang, and Hsin-Hsi Chen. 2020. Heterogeneous recycle generation for Chinese grammatical error correction. In *Proceedings of*

the 28th International Conference on Computational Linguistics, pages 2191–2201, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Jianshu Ji, Qinlong Wang, Kristina Toutanova, Yongen Gong, Steven Truong, and Jianfeng Gao. 2017. A nested attention neural hybrid model for grammatical error correction. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 753–762, Vancouver, Canada. 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. 2020a. 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.

Masahiro Kaneko, Masato Mita, Shun Kiyono, Jun Suzuki, and Kentaro Inui. 2020b. 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.

Sudhanshu Kasewa, Pontus Stenetorp, and Sebastian Riedel. 2018. Wronging a right: Generating better errors to improve grammatical error detection. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4977–4983, Brussels, Belgium. 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.

Philipp Koehn. 2005. Europarl: A parallel corpus for statistical machine translation. In *Proceedings of Machine Translation Summit X: Papers*, pages 79–86, Phuket, Thailand.

Moshe Koppel and Noam Ordan. 2011. Translationese and its dialects. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 1318–1326, Portland, Oregon, USA. Association for Computational Linguistics.

David Kurokawa, Cyril Goutte, and Pierre Isabelle. 2009. Automatic detection of translated text and its impact on machine translation. In *Proceedings of Machine Translation Summit XII: Papers*, Ottawa, Canada.

Jared Lichtarge, Chris Alberti, Shankar Kumar, Noam Shazeer, Niki Parmar, and Simon Tong. 2019. Corpora generation for grammatical error correction. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 3291–3301, Minneapolis, Minnesota. Association for Computational Linguistics.

Nitin Madnani, Joel Tetreault, and Martin Chodorow. 2012. Exploring grammatical error correction with not-so-crummy machine translation. In *Proceedings of the Seventh Workshop on Building Educational Applications Using NLP*, NAACL HLT '12, page 44–53, USA. Association for Computational Linguistics.

Anthony McEnery and Zhonghua Xiao. 2004. The Lancaster corpus of Mandarin Chinese: A corpus for monolingual and contrastive language study. In *Proceedings of the Fourth International Conference on Language Resources and Evaluation (LREC'04)*, Lisbon, Portugal. European Language Resources Association (ELRA).

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

Jakub Náplava and Milan Straka. 2019. Grammatical error correction in low-resource scenarios. In *Proceedings of the 5th Workshop on Noisy User-generated Text (W-NUT 2019)*, pages 346–356, Hong Kong, China. Association for Computational Linguistics.

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.

Nathan Ng, Kyra Yee, Alexei Baevski, Myle Ott, Michael Auli, and Sergey Edunov. 2019. Facebook FAIR's WMT19 news translation task submission. In *Proceedings of the Fourth Conference on Machine Translation (Volume 2: Shared Task Papers, Day 1)*, pages 314–319, Florence, Italy. Association for Computational Linguistics.

Maeve Olohan. 2002. Leave it out! using a comparable corpus to investigate aspects of explicitation in translation. *Cadernos de Tradução*, 1(9):153–169.

Myle Ott, Sergey Edunov, Alexei Baevski, Angela Fan, Sam Gross, Nathan Ng, David Grangier, and Michael Auli. 2019. fairseq: A fast, extensible toolkit for sequence modeling. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, pages 48–53, Minneapolis, Minnesota. Association for Computational Linguistics.

Myle Ott, Sergey Edunov, David Grangier, and Michael Auli. 2018. Scaling neural machine translation. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 1–9, Brussels, Belgium. 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 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*. Association for Computational Linguistics.

Thomas Payne. 2006. *Exploring language structure: A student's guide*. Cambridge University Press.

Ella Rabinovich, Sergiu Nisioi, Noam Ordan, and Shuly Wintner. 2016. On the similarities between native, non-native and translated texts. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1870–1881, Berlin, Germany. Association for Computational Linguistics.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2019. Exploring the limits of transfer learning with a unified text-to-text transformer. In *arXiv preprint arXiv:1910.10683*.

Marek Rei, Mariano Felice, Zheng Yuan, and Ted Briscoe. 2017. Artificial error generation with machine translation and syntactic patterns. In *arXiv preprint arXiv:1707.05236*.

Parker Riley, Isaac Caswell, Markus Freitag, and David Grangier. 2020. Translationese as a language in "multilingual" NMT. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7737–7746, Online. Association for Computational Linguistics.

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.

Alla Rozovskaya and Dan Roth. 2010. Generating confusion sets for context-sensitive error correction. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 961–970, Cambridge, MA. Association for Computational Linguistics.

Alla Rozovskaya and Dan Roth. 2019. Grammar error correction in morphologically rich languages: The case of Russian. *Transactions of the Association for Computational Linguistics*, 7:1–17.

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.

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.

Michael Stubbs. 1996. *Text and corpus analysis : computer-assisted studies of language and culture*. Language in society 23. Blackwell, Oxford.

Wenchao Su and Defeng Li. 2016. Corpus-Based Studies of Translational Chinese in English–Chinese Translation (2015). Richard Xiao and Xianyao Hu. *Digital Scholarship in the Humanities*, 31(3):516–519.

Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. 2014. Sequence to sequence learning with neural networks. In *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada*, pages 3104–3112.

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.

Gideon Toury. 1995. *Descriptive translation studies and beyond*, volume 4. J. Benjamins Amsterdam.

Ashish Vaswani, Samy Bengio, Eugene Brevdo, Francois Chollet, Aidan Gomez, Stephan Gouws, Llion Jones, Łukasz Kaiser, Nal Kalchbrenner, Niki Parmar, Ryan Sepassi, Noam Shazeer, and Jakob Uszkoreit. 2018. Tensor2Tensor for neural machine translation. In *Proceedings of the 13th Conference of the Association for Machine Translation in the Americas (Volume 1: Research Track)*, pages 193–199, Boston, MA. Association for Machine Translation in the Americas.

Shuo Wang, Zhaopeng Tu, Shuming Shi, and Yang Liu. 2020. On the inference calibration of neural machine translation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3070–3079, Online. Association for Computational Linguistics.

Shuo Wang, Zhaopeng Tu, Zhixing Tan, Shuming Shi, Maosong Sun, and Yang Liu. 2021. On the language coverage bias for neural machine translation. In *arXiv preprint arXiv:2106.03297*.

Felix Wu, Angela Fan, Alexei Baevski, Yann N. Dauphin, and Michael Auli. 2019. Pay less attention with lightweight and dynamic convolutions. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.

Richard Xiao. 2010. How different is translated chinese from native chinese?: A corpus-based study of translation universals. *International Journal of Corpus Linguistics*, 15(1):5–35.

Richard Xiao, Lianzhen He, and Ming Yue. 2008. The zju corpus of translational chinese (zctc). Text Corpus.

Ziang Xie, Guillaume Genthial, Stanley Xie, Andrew Ng, and Dan Jurafsky. 2018. Noising and denoising natural language: Diverse backtranslation for grammar correction. 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 619–628, New Orleans, Louisiana. Association for Computational Linguistics.

Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. 2020. mt5: A massively multilingual pre-trained text-to-text transformer. In *arXiv preprint arXiv:2010.11934*.

Helen Yannakoudakis, Øistein E. Andersen, Ardeshir Geranpayeh, Ted Briscoe, and Diane Nicholls. 2018. Developing an automated writing placement system for esl learners. *Applied Measurement in Education*, 31:251–267.

Helen Yannakoudakis, Ted Briscoe, and Ben Medlock. 2011. A new dataset and method for automatically grading esol texts. 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)*. Association for 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.

Mike Zhang and Antonio Toral. 2019. The effect of translationese in machine translation test sets. In *Proceedings of the Fourth Conference on Machine Translation (Volume 1: Research Papers)*, pages 73–81, Florence, Italy. Association for Computational Linguistics.

Wei Zhao, Liang Wang, Kewei Shen, Ruoyu Jia, and Jingming Liu. 2019. Improving grammatical error correction via pre-training a copy-augmented architecture with unlabeled data. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 156–165, Minneapolis, Minnesota. Association for Computational Linguistics.

Yuanyuan Zhao, Nan Jiang, Weiwei Sun, and Xiaojun Wan. 2018. Overview of the nlpcc 2018 shared task: Grammatical error correction. In *Natural Language Processing and Chinese Computing*, pages 439–445, Cham. Springer International Publishing.

Zewei Zhao and Houfeng Wang. 2020. Maskgec: Improving neural grammatical error correction via dynamic masking. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 1226–1233. Association for the Advancement of Artificial Intelligence (AAAI).

Wangchunshu Zhou, Tao Ge, Chang Mu, Ke Xu, Furu Wei, and Ming Zhou. 2020. Improving grammatical error correction with machine translation pairs. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 318–328, Online. Association for Computational Linguistics.

Michał Ziemski, Marcin Junczys-Dowmunt, and Bruno Pouliquen. 2016. The United Nations parallel corpus v1.0. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 3530–3534, Portorož, Slovenia. European Language Resources Association (ELRA).

A Appendix

A.1 Details of Quantifying Data Properties

One hypothesis is that the distribution of GEC data is similar to that of translationese. To verify our hypothesis, we follow the quantifying method proposed by Rabinovich et al. (2016) and Su and Li (2016) to explore the linguistic properties of the English and Chinese GEC data. If the statistical results are close, the data are similar in terms of different linguistic properties.

Data For the English data, we use the native texts and translationese released by the European Parliament Proceedings (Koehn, 2005). Additionally, we combine the native texts with WMT17 News Crawl monolingual data as the final native data. For the Chinese data, we use Lancaster Corpus of Mandarin Chinese (LCMC) (McEnery and Xiao, 2004) and People's Daily data as native language data. The ZJU Corpus of Translational Chinese (ZCTC) (Xiao et al., 2008) is used as the translationese. The English and Chinese GEC training data keep the same setting as mentioned in Section 5. For all the data types, we report normalized statistical results measured on 780k and 800k tokens for English and Chinese language, respectively.

Lexical richness Lexical richness is measured by the type-token ratio (TTR). Stubbs (1996) and Xiao (2010) point out that the lexical richness of native texts is larger than translationese in both English and Chinese. Our results show the same TTR trend as the result reported by Rabinovich et al. (2016) and Su and Li (2016).

Cohesive markers Connectives, which illustrate the logical relationships in sentence structure (Koppel and Ordan, 2011) and (Su and Li, 2016), are more commonly used in translationese compared to native texts. To verify this property, we collect about 116 cohesive markers for English and 150 for Chinese. The measurement is calculating the frequency of these cohesive markers that appeared in the four data types. The results show that the connective frequency of translationese and GEC data are higher than the native texts in both English and Chinese languages.

Collocations Native language speakers tend to use common and frequent collocations (Britt et al., 2015). We collect about 8,300 commonly used collocations for English and 6,100 for Chinese. The measurement is computing the frequency of these collocations used in the four data types. The results show that our native language data and GEC data have a similar frequency distribution compared to the results reported by the previous study (Rabinovich et al., 2016) and (Su and Li, 2016).

Pronouns The usage of pronouns is different in Chinese and English. For English, translators prefer to write the actual nouns rather than pronouns that reflect the principle of explicitation (Olohan, 2002). However, Chinese translators are often influenced by the source text and directly translate the pronoun (Su and Li, 2016). The measurement is the frequency of the pronouns in the four data types. The results show that the trends in Chinese are consistent with the result mentioned by Su and Li (2016). For English, the GEC data has more pronouns compared to our own native data, such as "I" and "you".

Content Words and Function Words We use the `Stanford POS tagger`[4] to annotate contents and function words for both English and Chinese. For content words, we calculate the frequency of adjectives, pronouns, nouns, and verbs in the four data types. For function words, we calculate the frequency of conjunctions, adverbs, determiners, and prepositions. The results show that translationese tends to use more function words to make the sentences simple and explicit (Su and Li, 2016). Besides, the frequency distribution in translationese is similar to GEC data.

A.2 Settings of BERT Classifier

The settings of hyper-parameters of the fine-tuning BERT classifiers are listed in Table 5.

Configurations	Values
Model Architecture	BERT (Devlin et al., 2019)
Max Input Length	128
Learning Rate	0.00002
Traning Epochs	2
Batch Size	32
Other Settings	Default

Table 5: Hyper-parameters for English, German, Russian, and Chinese BERT classifiers.

A.3 Case Study for the Identified Texts

We present the examples of English native texts and translationese distinguished from the UN corpus

[4] https://nlp.stanford.edu/software/tagger.shtml

Native	He would continue consultations in 2008 with a view to holding the next Conference session in a new region, to reinforce Member States' ownership of the Organization.
Native	She urged States to bear in mind the importance of ensuring and maintaining the contextual space for the activities of human rights defenders, including the right to peaceful assembly, in combination with the rights entailed in relation to freedom of expression and association.
Translationese	Ms. Andersen (Denmark) said that sexual harassment in the workplace was strictly prohibited and that protection was available through the Gender Equality Board and the courts.
Translationese	An appropriate legal framework would ensure the validity and enforceability of electronic transactions in all circumstances and create certainty in such an important area of law.
Non-native	Because some of my classmates make great progress in the exam and they catch up with me and some of them even surpass me.
Non-native	The students are so nice and obedient, which is very good for me because I am a beginner.

Table 6: Examples of the native texts and tanslationese distinguished by the BERT-based pre-trained classifier. **Native** (**Translationese**) refers to the examples of native (tanslationese) texts. **Non-native** refers to the examples of GEC train data. The words with the color red represent the characteristics of native texts. The words with the color blue resemble the characteristics of the second language learners.

Deletion	Insertion	Replacement	F_{05}
0.1	0.1	0.1	55.82
0.1	0.1	0.2	55.87
0.1	0.2	0.3	56.18
0.05	0.1	0.2	**56.23**
0.05	0.1	0.3	56.21
0.05	0.2	0.4	56.15

Table 7: $F_{0.5}$ scores of the probabilities of translationese corruption with deletion, insertion and substitution for different groups. **Bold** value indicates the best result.

Lan.	Corpus	Train	Dev	Test
DE	cLang8-de	0.11M	-	-
DE	Falko-MERLIN	-	2.503	2,337
RU	cLang8-ru	45K	-	-
RU	RULEC-GEC	-	2,500	5,000

Table 8: Statistics of the data sets for German and Russian GEC models training and finetuning

by our proposed BERT-based classifier in Table 6. It can be seen that the native texts contain collocations (idioms) like "with a few to", and "bear in mind", while translationese and the second language learners (non-native) data hardly contain them. The translationese and non-native texts contain more cohesive markers like "and" and "because" than native texts. In addition, native texts like to use pronouns, but translationese and the second language learners' data tend to give the specific content which indicates the characteristic of explicitation. Overall, the examples show that translationese resembles the second language learners' data in many aspects.

A.4 Ablation study of the corruption probabilities

Table 7 presents six groups of different translationese corruption probabilities with deletion, insertion and substitution. We can see that the choice of different corruption probabilities does not make a big difference in the results. We choose the probabilities of 0.05, 0.1, 0.2 in our experiments as it works best of the six.

A.5 Settings of GEC Models Training

The hyper-parameters settings of the training Transformer GEC models are listed in Table 9.

A.6 Settings of (m)T5 Fine-tuning

Table 10 presents the hyper-parameters for fine-tuning T5/mT5 GEC models.

A.7 Results for German and Russian Trained on cLang-8 Datasets

Table 8 present the statistics of the cLang8 data used for training and finetuning German and Russian GEC tasks based on the Transformer-base model and mT5 large pre-trained model. For training English and Chinese GEC models on the Transformer-base model, we use the same data presented in Table 1 in the text. Table 11 shows that the model augmented with translationese (i.e.,+TRANS) outperforms the BASE and +NATIVE method on TRANSF. for English, Chinese, German and Russian GEC benchmarks both on Transformer and MT5 LARGE models. Even though our results are not reached the strong baselines LRGEC (Náplava and Straka, 2019) for Ger-

Config.	English GEC Model	Chinese GEC Model	German GEC Model	Russian GEC Model
Model Arch.	Transformer-base	Transformer-base	Transformer-base	Transformer-base
Optimizer	Adam	Adam	Adam	Adam
Adam-Betas	$\beta_1 = 0.9, \beta_2 = 0.98$	$\beta_1 = 0.9, \beta_2 = 0.998$	$\beta_1 = 0.9, \beta_2 = 0.98$	$\beta_1 = 0.9, \beta_2 = 0.98$
LR	0.0007	0.0007	0.0005	0.0005
Dorpout	0.3	0.2	0.3	0.3
Att. Drop.	0.1	0.1	0.1	0.1
Act. Drop.	0.1	0.1	0.1	0.1
Batch Size	16,384	8,192	8,192	4,096
Update Freq	2	2	1	1
Beam Size	5	12	5	5

Table 9: Hyper-parameters for training English, Chinese, German and Russian GEC models. Model Arch. refers to model architecture, LR is learning rate, Att. Drop. means attention dropout, Atc. Drop. means activation dropout.

Config.	English GEC Model	Chinese GEC Model	German GEC Model	Russian GEC Model
Model Arch.	T5-Large	mT5-Large	mT5-Large	mT5-Large
Optimizer	Adafactor	Adafactor	Adafactor	Adafactor
LR	0.001	0.0007	0.0007	0.001
Batch Size	2,048	1,536	1,536	1,024
Update Freq	128	128	128	128
Beam Size	5	5	5	5

Table 10: Hyper-parameters for fine-tuning English, Chinese, German and Russian GEC models. Model Arch. refers to model architecture. LR denotes the learning rate.

man and Russian, our results also sufficiently confirm the effectiveness of our approach compared to the GEC models trained on the same training data and model settings (Rothe et al., 2021). The training settings for the aforementioned models are presented in A.5 and A.6.

A.8 Case Study for GEC Models Outputs

Table 12 shows some outputs generated by native/translationese GEC model. By taking English as an example, the translationese GEC model corrects ungrammatical sentences better than native GEC model. It indicates that using translationese as input for GEC data augmentation can improve performance.

A.9 Results on the BEA19 English Test

Table 13 shows that the model augmented with translationese (i.e.,+TRANS) outperforms the other settings on BEA19 W&I non-native test and BEA19-ALL test. However, the +NATIVE method is better than others on BEA19 LOCNESS native test. After borrowing knowledge from the T5 pre-trained model, the performance still remains consistent and achieves promising results. Overall, the results sufficiently confirm the effectiveness of utilizing similar style texts as input for data augmentation.

A.10 Details of Linguistic Properties Settings

Word frequency and word position reflect the performance of GEC systems from the perspective of word-level accuracy and sentence structure, respectively. We use the `compare-MT`[5] toolkit to compare the outputs of BASE, NATIVE, MIX and TRANS. GEC models by F-measure. Taking the result of BASE model as a baseline, we report the improvements of each GEC model.

Word Frequency: We count the word frequencies of English and Chinese GEC on the target training sets, dividing their tokens into three categories according to their frequency. We follow Wang et al. (2020) to select the most 3,000 frequent tokens into the *Frequent* bucket, the most 3,001-12,000 into *Medium* bucket, and the others into the *Rare* bucket for English and Chinese.

Position: From the perspective of sentence structure, the behavior of GEC models may be different at different positions of the sentence. We divide the sentences into three buckets that have equal length and categorize the token into three types based on which bucket they belong to, which are *Left*, *Middle* and *Right*. Specifically, it firstly gives every token a number in each sentence according to the formula: $P/N - 1$, N is the length of the sentence.

[5] https://github.com/neulab/compare-mt

Model	Method	EN (CoNLL14)			ZH (NPLCC18)			DE (Falko-MERL.)			RU (RULEC-GEC)		
		Pre.	Rec.	$F_{0.5}$	Pre.	Rec.	$F_{0.5}$	Pre.	Rec.	$F_{0.5}$	Pre.	Rec.	$F_{0.5}$
MaskGEC	-	-	-	-	44.4	22.2	37.0	-	-	-	-	-	-
TagGEC	-	72.8	49.5	66.6	-	-	-	-	-	-	-	-	-
LRGEC	-	-	-	63.4	-	-	-	78.2	59.9	73.7	63.3	27.5	50.2
(m)T5 large	-	-	-	66.0	-	-	-	-	-	70.1	-	-	27.6
(m)T5 xxl	-	-	-	68.8	-	-	-	-	-	74.8	-	-	43.5
Transf.	Base	60.1	36.6	53.3	31.2	20.2	28.1	58.8	34.3	51.5	3.6	**1.9**	3.1
	+Native	63.0	37.2	55.3	34.5	22.2	31.1	62.7	31.8	52.5	5.8	1.4	3.6
	+Mix	63.6	**37.5**	55.8	34.5	23.0	31.4	62.9	32.3	52.9	5.5	1.8	3.9
	+Trans.	**64.2**	37.5	**56.2**†	**35.6**	**23.6**	**32.3**†	**63.1**	**32.9**	**53.3**†	**6.2**	1.8	**4.2**†
(m)T5 large	Base	71.8	51.4	66.5	41.5	25.8	37.0	75.4	55.1	70.2	42.6	17.9	33.4
	+Native	73.2	51.4	67.5	43.6	24.6	37.8	75.9	55.9	70.8	43.6	18.4	34.2
	+Mix	73.8	51.2	67.8	43.1	**26.5**	38.3	**76.0**	57.6	71.4	44.9	19.3	35.5
	+Trans.	**74.7**	**51.6**	**68.6**†	**45.2**	24.5	**38.7**†	75.8	**58.9**	**71.7**†	**45.1**	**20.1**	**36.1**†

Table 11: Results on CoNLL14 English, NLPCC18 Chinese, Falko-MERLIN German, and RULEC-GEC Russian GEC tasks, which trained and funetuned on the cLang-8 GEC training data for Transformer and (m)T5 large models. mT5 large results indicate the fine-tuned T5 large models with the same cLang8 GEC data, which was reported by Rothe et al. (2021). Statistically significant improvements over +Native method are reported using P_value, $^{\dagger}p < 0.01$.

Src	Do one who suffered from this disease keep it a secret of infrom their relatives ?
Ref	Does **someone** who **suffers** from this disease keep it a secret **or** inform their relatives ?
Native-gen	Does one who suffered from this disease keep it a secret from their relatives ?
Trans.-gen	Does **anyone** who **suffers** from this disease keep it a secret from their relatives ?
Src	And both are not what we want since most of us just want to live as normal people .
Ref	And both are not what we want **, since** most of us just want to live as normal people .
Native-gen	But both are not what we want since most of us just want to live as normal people .
Trans.-gen	And both are not what we want **, since** most of us just want to live as normal people .

Table 12: Examples of outputs generated by Native/Translationese GEC model. **Src** is the source ungrammatical sentence, **Ref** is the target corrected sentence. **Native**-gen (**Trans.**-gen) refers to the native (tanslationese) GEC model outputs. The words with the color red are the error parts and the **bold** words indicate the corrected version. The translationese GEC model corrects ungrammatical sentences better.

Model	Method	BEA19 W&I test			BEA19 LOCNESS test			BEA19-ALL test		
		Pre.	Rec.	$F_{0.5}$	Pre.	Rec.	$F_{0.5}$	Pre.	Rec.	$F_{0.5}$
T5 large	-	-	-	-	-	-	-	-	-	72.1
Transf.	Base	63.0	49.2	59.7	45.6	**52.9**	46.9	60.9	48.3	57.9
	+Native	67.6	50.6	63.3	48.6	49.4	**48.8**	64.8	48.9	60.8
	+Mix.	67.7	50.5	63.4	**48.7**	46.9	48.3	65.1	49.0	61.1
	+Trans.	**68.1**	**50.8**	**63.8**	48.2	48.0	48.3	**65.5**	**49.7**	**61.6**
T5 large	Base	74.6	66.2	72.8	71.1	77.3	72.3	73.4	**67.0**	72.0
	+Native	76.5	**66.6**	74.3	**76.8**	74.5	**76.3**	75.1	66.1	73.1
	+Mix.	**77.2**	65.5	74.5	75.4	**76.9**	75.7	**76.0**	65.4	73.6
	+Trans.	77.1	66.2	**74.6**	75.0	76.8	75.4	75.8	66.0	**73.6**

Table 13: Results on the BEA19 test set. BEA19 W&I is A,B,C-level non-native test sets, and BEA19 LOCNESS refers to the BEA19 native test set. BEA19 ALL is the full BEA19 benchmark. T5 large results use cLang-8 data fine-tuned on the T5-large pre-trained model, which was reported by Rothe et al. (2021).

p is the position of each token, $p \in [0, N-1]$. Then, we set the threshold values, if the number of tokens $< 1/3$, it belongs to the left bucket; if the number of tokens $> 2/3$, it belongs to the right bucket, and the others belong to the middle bucket.