Comprehensive Evaluation of Translation Error Correction Models

Masatoshi Otake and **Yusuke Miyao**
The University of Tokyo
{otake,yusuke}@is.s.u-tokyo.ac.jp

Abstract

In this study, we examine the grammatical error correction capabilities of **Translation Error Correction (TEC)** models and investigate a technique for generating pseudo TEC data by injecting pseudo grammatical errors into a bilingual corpus. Translation Error Correction (TEC) is the field of automatically correcting errors in translated texts. Despite extensive research conducted in the field of **Grammatical Error Correction (GEC)**, where computers are utilized to correct grammatical inaccuracies, studies dedicated to TEC remain remarkably limited. Previous research has demonstrated the potential of learner-oriented TEC; however, TEC's ability to correct grammatical errors remains uncertain due to limited investigation. To address this, we apply a range of methods commonly employed in GEC—including the use of pseudo errors and pre-trained models—and conduct a comprehensive evaluation. Also, we propose a new method to create a pseudo dataset of TEC. Our results show that TEC is stronger than GEC in the general experiment settings and that our pseudo data is effective.

1 Introduction

The field of grammatical error correction (GEC), which involves computer systems correcting textual inaccuracies, and its applications, such as Grammarly,[1] are rapidly evolving, with a growing demand in real-world scenarios. However, students who study English writing cannot be fully helped by computers yet. One potential reason is that the current error correction systems cannot rectify complex errors. For instance, Cao et al. (2018) showed the difficulty when attempting to correct a sentence like "I am leaving in Tokyo", due to the inherent ambiguity in deciding whether to change "leaving" to "living", or "in" to "for". Current error correction systems cannot correct ambiguous errors. To address this, we focus on the fact that second-language learners of English start from translation from their native language to English. (In this paper, we focus on native Japanese speakers and English learners.) In this context, error correction systems may be able to perform better corrections by referencing the original native language text (Figure 1). In this paper, we refer to the task of correcting errors in translation as "Translation Error Correction (TEC)" (Lin et al., 2022). The term 'TEC' was defined by Lin et al. (2022) as a task to correct the errors of professional translators, but in our current study, we focus on rectifying the errors made by learners.

The initiation of learner-oriented TEC research can be attributed to Cao et al. (2018), who referred to it as multi-source grammatical error correction. They created a TEC dataset from a GEC dataset and showed the potential of TEC models ability to correct grammatical errors. Although their study yielded some promising results, it did not provide a comprehensive understanding of TEC grammatical correction ability because of limited experiments. The primary issue is that their study has not adhered to these established standard settings, while GEC has been extensively researched with sophisticated datasets, methodologies, and evaluation strategies. They used a part of the Lang-8 learner corpus (Mizumoto et al., 2011) for evaluation, but this made replication difficult and comparison with GEC challenging. Additionally, strategies that proved effective in GEC, such as the use of pseudo data (Xie et al., 2018; Ge et al., 2018; Lichtarge et al., 2019; Kiyono et al., 2019) and pre-trained models (Kaneko et al., 2020; Katsumata and Komachi, 2020), were not tested. This leaves the actual grammatical correction capability of TEC models largely unexplored.

In order to unravel *the grammatical correction capability of TEC models*, we conducted exhaus-

[1] https://grammarly.com/

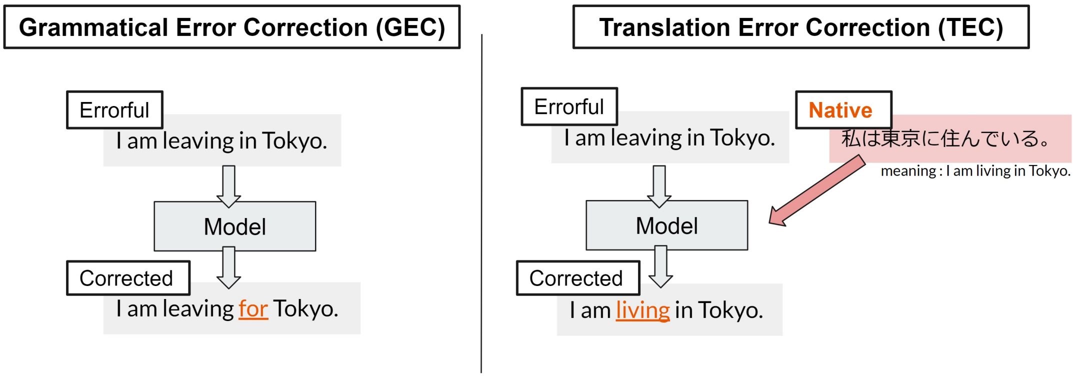

Figure 1: Comparison between Grammatical Error Correction (GEC) and Translation Error Correction (TEC). The example, TEC can solve grammatical errors with ambiguity.

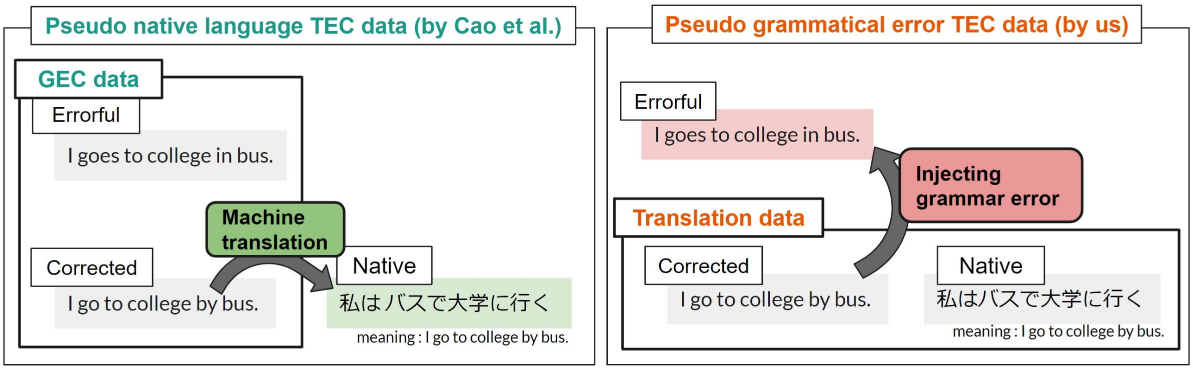
Pseudo native language TEC data (by Cao et al.)
GEC data
Errorful
I goes to college in bus.
Machine translation
Corrected
I go to college by bus.
Native
私はバスで大学に行く
meaning : I go to college by bus.
Pseudo grammatical error TEC data (by us)
Errorful
I goes to college in bus.
Injecting grammar error
Translation data
Corrected
I go to college by bus.
Native
私はバスで大学に行く
meaning : I go to college by bus.

Figure 2: Dataset creation of TEC. The left figure describes the pseudo native language TEC dataset proposed by Cao et al. (2018). The right one describes the pseudo grammatical error TEC dataset proposed by us.

tive experiments using the standard settings of GEC. For example, as test datasets, we employed BEA-2019-dev (Granger, 1998; Yannakoudakis et al., 2011), CoNLL-2014 (Ng et al., 2014), and JFLEG (Napoles et al., 2017), which are commonly used in GEC assessments. Additionally, we carried out an inclusive examination of typical approaches in GEC model construction, such as pseudo datasets and pre-trained models, thereby thoroughly examining TEC effectiveness. Further, we propose a method for creating large-scale pseudo data for TEC using parallel corpora and pseudo grammatical error generation methods because pseudo data from GEC cannot be directly applied to TEC. By employing such standard GEC methodologies, we could compare TEC with various GEC research. Our results demonstrated that the ability of TEC models to correct grammatical errors surpasses GEC in the general experiment settings and that our pseudo data is consistently effective on TEC.

2 Related work

2.1 Prior research by Cao et al. (2018)

As highlighted in Section 1, the direct precedent for this study is the research by Cao et al. (2018). TEC requires three types of data: Native language sentences, errorful English sentences, and their corrected English sentences (Figure 1). However, there is no gold dataset that has all three of these data sets. Cao et al. (2018) proposed a method for creating a TEC dataset. They first machine-translated corrected English texts from the GEC corpus into native language sentences (i.e. Japanese sentences). This process is illustrated in Figure 2. GEC corpus is constructed of errorful sentences and corrected sentences, and the native language sentences are translated by machine translation. The GEC corpus, Lang-8 (Mizumoto et al., 2011), was used as the original source of the TEC dataset, and then it was divided into training, development, and evaluation datasets. Furthermore, Cao et al. (2018) proposed an LSTM-based model for TEC that incorporates two encoders, each separately processing the native and errorful sentences, and compared GEC with TEC. The evaluation metric was the GLEU score (Napoles et al., 2015), a variant of the BLEU score adapted for GEC. They demonstrated that using native sentences resulted in higher GLEU scores than using only errorful sentences.

While Cao et al. (2018) was quite pioneering and remarkable, it encountered certain issues. Two of the main problems were its non-standard approach to measuring the ability to correct grammatical errors and the limited scope of model construction. Firstly, they utilized the Lang-8 (Mizumoto et al., 2011) corpus for evaluation, but it was not standard and caused some problems. Since Lang-8 data was collected from a social networking service (SNS) for error correction, it is considered noisy and may not be suitable for evaluation purposes. It has not been used in GEC research as an evaluation dataset, complicating any direct comparisons with GEC studies. Furthermore, since the Lang-8 corpus does not define specific evaluation data and Cao et al. (2018)'s partitioning is not completely defined, other researchers face challenges reproducing the experiments. Secondly, although GEC research has demonstrated that some methods are quite effective, such as pre-trained models and pseudo data, their research did not test them. In this study, we aim to assess the grammatical error correction capability of TEC by adopting standard experimental settings from GEC and conducting comprehensive experiments.

2.2 Other TEC research

Professional translation error correction TEC has also been studied as a task for correcting errors made by professional translators (Lin et al., 2022). Lin et al. (2022) were the first to introduce the term TEC task. They constructed a TEC corpus of professional translators and compared the automatic post-editing (APE) model with the model they trained for TEC using this corpus. APE is the task of fixing machine translation output, and already many APE models are researched and constructed. Lin et al. (2022) argued that TEC, which corrects human translations, and APE, which corrects machine translations, might exhibit different tendencies. The result showed that TEC is better than APE, and they asserted that a model for TEC, rather than APE, should be used for professional translators' corrections.

Our task focuses on errors made by learners while they focus on errors made by professional translators. The nature of errors in these two tasks can be significantly different. For example, in professional translation, the use of specialized terms and correct style is demanded. In the DIGITAL-OCEAN (DO) corpus they created, such errors related to specialized terms and style comprise 40%. On the other hand, our learner-oriented TEC focuses solely on grammatical errors. Moreover, learner texts often tend to revolve around daily or social themes, while the texts for professional translators are specialized.

Error correction as scoring Short Answer Scoring (SAS) is a task to automatically grade short answers. Kikuchi et al. (2021) dedicated their efforts to grading learner translations using machines. Although their task is entirely different from ours, they and we share the common point of leveraging machines to tackle translations from second language learners. While our work aims to generate correct sentences from learners' texts, their approach is targeted to provide scores and feedback across various categories like tense and conjunctions.

2.3 Method for generating pseudo grammatical errors

We generate a TEC pseudo dataset because the methods to generate pseudo grammatical errors mechanically have been extensively studied as part of GEC, and these methods can extend the datasets (Xie et al., 2018; Ge et al., 2018; Lichtarge et al., 2019; Kiyono et al., 2019). For example, Kiyono et al. (2019) showed that using pseudo data leads to an increase of 2-3 points in the $F_{0.5}$ score of BEA-2019-dev. Among these researches, the error generation method using back-translation has been reported to be effective (Xie et al., 2018; Kiyono et al., 2019), which we adopted for this study. In the context of GEC, back-translation refers to constructing a model that uses GEC corpora to input corrected sentences and output erroneous ones, then applying it to a monolingual corpus to generate synthetic error data.

2.4 Pre-trained language models for GEC

We also apply a pre-trained language model to our TEC experiments because the method of fine-tuning pre-trained language models has been reported to be effective in GEC (Kaneko et al., 2020; Katsumata and Komachi, 2020). For example, Kaneko et al. (2020) showed that using the pre-trained encoder model BERT (Devlin et al., 2019) caused to increase of about 3 to 5 points in the $F_{0.5}$ score on BEA-2019-test. Katsumata and Komachi (2020) showed that GEC models can get enough scores by utilizing BART (Lewis et al., 2020).

3 Proposed method

We aim to understand the ability of TEC models to correct grammatical errors comprehensively. Therefore, we apply the *standard experimental setup of GEC to TEC* and utilize experimental methods proven effective in GEC within the TEC context because GEC has already been extensively studied. Furthermore, we propose a method for *generating large-scale pseudo TEC data* to employ pseudo data in TEC because pseudo data has proved quite valuable in GEC.

Pseudo TEC data In our experiment, we employ two types of pseudo TEC data, which can be confusing. We refer to the reproduced TEC datasets originally proposed by Cao et al. (2018) as *pseudo native language TEC data* and the data proposed by us as *pseudo grammatical error TEC data*. Figure 2 explains the difference between pseudo native language TEC data and pseudo grammatical error TEC data. Pseudo native language TEC data is created from GEC dataset and machine translation, and pseudo grammatical error TEC data is created from the translation dataset and automatic grammatical error injection.

We use pseudo grammatical error TEC data as pre-training data and use pseudo native language TEC data as fine-tuning and evaluation data because, in TEC, the quality of grammatical errors is more important than that of native sentences.

3.1 Validation of TEC in grammatical error correction

For standard experiments based on GEC, it is believed to be crucial to standardize the source datasets, evaluation, and methodology.

Cao et al. (2018) developed pseudo native language TEC data by using source GEC datasets and machine translation (Figure 2). Cao et al. (2018) machine translated the correct sentences of a GEC dataset and created TEC data. We also develop TEC data in the same way.

Dataset	Sentences
Lang-8	1,037,561
NUCLE	57,151
FCE	33,236
W&I+LOCNESS	34,304
Total	1,162,252
Corrected	564,688
Our pseudo data	8,300,633

Table 1: The sentence pairs of training datasets (Corrected) and those of pseudo pre-training dataset. Our 'pseudo' means 'pseudo grammatical error'. We utilize only those sentence pairs with corrections (Kaneko et al. (2020)).

Dataset	Metric	Sentences
BEA-2019 dev	$F_{0.5}$(ERRANT)	4384
CoNLL-2014	$F_{0.5}$(MaxMatch)	1312
JFLEG	GLEU	747

Table 2: The sentence pairs and metrics of evaluation datasets.

Source datasets As the source dataset of TEC, we mainly utilize the BEA-2019 shared task datasets (Bryant et al., 2019; Granger, 1998; Mizumoto et al., 2011; Tajiri et al., 2012; Yannakoudakis et al., 2011; Dahlmeier and Ng, 2012) because it is standard in GEC. We make a training dataset and development TEC dataset from the BEA-2019 training dataset and the BEA-2019 development dataset. This experiment setting enables comparison with other studies in GEC. The number of sentence pairs is shown in Table 1. As the training data, we utilize only those sentence pairs that had corrections, amounting to a total of 564,688 sentence pairs. (Kaneko et al. (2020))

As evaluation datasets, we use BEA-2019-dev (Granger, 1998; Yannakoudakis et al., 2011), CoNLL-2014 (Ng et al., 2014) and JFLEG (Napoles et al., 2017), which are datasets with publicly available corrected sentences and are frequently used as evaluation data in GEC. In GEC, it is crucial to evaluate the model using multiple datasets (Mita et al., 2019). Therefore, we apply not only BEA-2019 but also JFLEG and CoNLL-2014. The number of sentence pairs in each dataset is shown in Table 2. The reason why we do not use BEA-2019-test is that it is not public. In order to create pseudo-native language TEC data, corrected sentences need to be public.

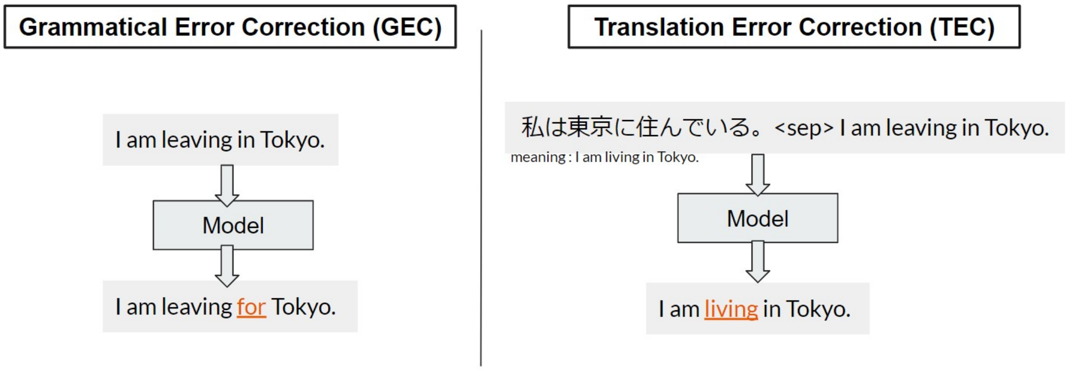
Grammatical Error Correction (GEC)
I am leaving in Tokyo.
Model
I am leaving for Tokyo.
Translation Error Correction (TEC)
私は東京に住んでいる。<sep> I am leaving in Tokyo.
meaning : I am living in Tokyo.
Model
I am living in Tokyo.

Figure 3: Comparison of input sentences of our experiments between Grammatical Error Correction (GEC) and Translation Error Correction (TEC).

Evaluation ERRANT (Bryant et al., 2017), MaxMatch (Dahlmeier and Ng, 2012), and GLEU (Napoles et al., 2015) were used for each evaluation dataset. For BEA-2019-dev evaluation, we use a distributed m2 file for evaluation.

Methodology We apply two approaches to TEC: the use of pre-trained language models and dataset augmentation with pseudo data. This is because it is shown that they are effective on GEC (Katsumata and Komachi, 2020; Rothe et al., 2021; Xie et al., 2018; Ge et al., 2018; Lichtarge et al., 2019; Kiyono et al., 2019). Regarding pseudo data, we propose a new method for generating pseudo data specifically tailored for TEC. This will be detailed in the following subsection. We utilize pre-trained language models and fine-tune them for TEC. Drawing on the GEC experiments conducted by Katsumata and Komachi (2020), we choose to use mBART. Since TEC requires two languages as input, it is necessary to use the pre-trained model that has been trained on a group of languages, including the two languages.

Furthermore, note that in this study, the native language sentences for TEC are created from the corrected sentences, which serve as answers. We conducted experiments with only native language sentences as input to verify whether there was any significant leakage of answers from the native language sentences.

3.2 Creation of pseudo grammatical error TEC data

As we mentioned in the last paragraph, we generate a pseudo TEC dataset and utilize it because pseudo datasets are helpful for GEC. We generate pseudo grammatical error TEC data by inserting pseudo grammatical errors into a parallel corpus and then evaluate the effectiveness of this data using pre-training (Figure 2).

We use JparaCrawl (Morishita et al., 2020) as the parallel corpus and employ back-translation (Xie et al., 2018) as the method to generate pseudo grammatical errors and thus create the pseudo grammar error TEC data. JparaCrawl is a large-scale Japanese-English parallel corpus created by crawling the web, and we used only sentences with a bleualign score of 0.75 or higher. The extracted data from JparaCrawl consists of 8,300,633 sentence pairs.

After pre-training with this pseudo grammatical error TEC data, we perform fine-tuning with pseudo native language TEC data to create the TEC model.

4 Experimental settings

4.1 Settings to create datasets

Pseudo native TEC data As is the same as Cao et al. (2018), we developed pseudo native TEC datasets by using GEC datasets and machine translation (The left side of Figure 2). We used DeepL[2], which was accessed in December 2022, to create the native (Japanese) sentences.

In TEC, there are two input sentences, thus, there are various methods for constructing input. In our experiments, the input sentences of TEC were created by simply concatenating the native (Japanese) sentences and error sentences, like '**私は東京に住んでいる。 <sep> I am leaving in Tokyo.**' (The right side of Figure 3). '**私は東京に住んでいる。**', which means '**I am living in Tokyo'** in English, is the native sentence, and '**I**

[2] https://www.deepl.com/

Params	Values
Epochs	60
Optimizer	adam
LR	1×10^{-4}
LR-scheduler	inverse_sqrt
Criterion	label_smoothed_cross_entropy
Dropout	0.3
Max tokens	4096

Table 3: Main hyperparameters to train models

am living in Tokyo.' is the errorful English sentence.

Pseudo grammatical error TEC data We create pseudo grammatical error TEC data by adding pseudo grammatical errors to a parallel corpus JparaCrawl (Morishita et al., 2020) for pre-training TEC. For injecting grammatical errors, we adopted back-translation. We used BART (Lewis et al., 2020) for the back-translation model and trained it with BEA-2019 training datasets. We employed noised back-translation (Xie et al., 2018), which adds noise during beam search, and the parameter β was set to 8.0 as is same with Koyama et al. (2021).

4.2 The settings of the models

As error correction models' architecture, we use a standard sequence-to-sequence Transformer architecture with 12 layers for the encoder and decoder. The pre-trained language model we used was mbart.cc25 (Liu et al., 2020). In addition, sentencepiece (Kudo and Richardson, 2018) was applied to all data, and for sentencepiece model, we used the tokenization models shared by Liu et al. (2020). The output was generated by performing a beam search with five beams and selecting the most probable output. Other detailed hyperparameters are shown in Table 3. We did four random seed experiments for each experimental setting.

5 Results

5.1 Validation of TEC in grammatical error correction

Table 4 presents the results of comparing TEC with GEC. This table is divided into six sections. The first four sections compare the grammatical error correction capabilities of GEC and TEC. The fifth section provides auxiliary experiments for discussing leakage issues, and the last section presents the results of previous GEC studies. In the first four sections, GEC and TEC are compared within each section. The abbreviation 'LM' signifies that the training started from a pre-trained language model, and 'pseudo' indicates whether pre-training was conducted using our pseudo grammatical error TEC data.

GEC vs. TEC In the baseline setting (the first section in the Table 4), the TEC model lags behind the GEC model by more than one point in two out of three experimental settings. However, in other settings (the second to fourth sections), TEC models show a comparable or better ability to correct grammatical errors than GEC models.

Effectiveness of pre-trained language models It is observed that pre-trained models contribute significantly to the results. This is already shown in GEC (Kaneko et al., 2020), and the same effect was found to hold true for TEC as well. When comparing the TEC baseline model (TEC: second row in the table) and the model utilizing the LM (TEC+LM: fourth row), we observe an improvement of over 10 points in BEA-2019 and CoNLL-2014, and an increase of more than 2 points in GLEU. Furthermore, consistently higher performance is seen in the model utilizing both our pseudo-data and pre-training models (TEC+LM+pseudo: eighth row) compared to the one using only the pseudo-data (TEC+pseudo: sixth row).

Comparison with previous work GEC results of Katsumata and Komachi (2020) have higher scores than our GEC results, probably because of the difference in the pre-trained model. Their model is constructed from only English, but mbart is constructed from 25 languages.

Leak check The fifth section presents an experiment wherein we input only Japanese sentences to produce corrected sentences, checking for any serious leaks in the answers. This experiment is necessary to verify that there is no serious leak because these native language sentences are made from answers, that is, corrected sentences. The scores in the fifth section significantly lag behind those in the first four sections, with most falling behind by more than 10 points. The low scores when only Japanese is used (ninth and tenth row) suggest that the leakage of correction information is unlikely to have occurred.

Method	TEC	LM	pseudo	BEA-2019-dev (ERRANT)			CoNLL-2014 (MaxMatch)			JFLEG
				P	R	$\mathbf{F_{0.5}}$	P	R	$\mathbf{F_{0.5}}$	**GLEU**
GEC				24.49	22.90	**24.16**	47.97	31.38	**43.38**	52.74
TEC	✓			22.31	29.37	23.43	44.44	37.22	42.76	**53.54**
GEC+LM		✓		44.04	31.80	40.65	60.87	41.01	55.38	**56.56**
TEC+LM	✓	✓		47.79	33.83	**43.90**	62.24	42.25	**56.72**	56.29
GEC+pseudo			✓	40.25	27.39	36.79	62.09	38.20	55.18	57.63
TEC+pseudo	✓		✓	39.59	38.27	**39.31**	59.92	47.01	**56.80**	**59.00**
GEC+LM+pseudo		✓	✓	47.18	30.65	42.56	68.32	41.39	**60.43**	59.35
TEC+LM+pseudo	✓	✓	✓	44.47	42.61	**44.07**	62.35	49.41	59.24	**60.03**
Translation	T			5.68	23.83	6.70	30.31	49.13	32.83	21.19
Translation+LM	T	✓		7.77	31.02	**9.14**	31.57	54.39	**34.46**	**26.24**
Kaneko et al. (2020)				-	-	-	59.2	31.2	50.2	52.7
Kaneko et al. (2020)		bert		-	-	-	63.6	33.0	53.6	54.4
Kaneko et al. (2020)		bert	✓	-	-	-	69.2	45.6	**62.6**	**61.3**
Katsumata (2020)		bart		-	-	-	69.3	45.0	**62.6**	57.3

Table 4: The results of our methods and previous work of GEC. TEC means using native language sentences, and the absence of a checkmark means GEC. LM means using a pre-trained language model, and pseudo means using a pseudo TEC (or GEC) dataset for pre-training. The top of the four groups is the comparison between GEC and TEC in each experiment setting. The fifth group is the experiment using only Japanese sentences to check the leak of the answers. The sixth group is the results of the previous work of GEC. Katsumata and Komachi (2020) and Kaneko et al. (2020) has only errorful input (GEC). Additionally, the pseudo dataset size of Kaneko et al. (2020) is about 70M, and ours is about 8M. **Bold** indicates the highest score of each group. Precision (P) and Recall (R) are metrics that need to be considered comprehensively and therefore are not emphasized in bold. Scores of our experiments are an average of 4 times experiments.

5.2 Effectiveness of our pseudo grammatical error TEC data

We compare the model pre-trained on our pseudo datasets with the model not pre-trained and show how effective our pseudo data is. The results are also shown in Table 4. The results demonstrate that the models pre-trained on our data consistently lead to higher scores compared to the models without pre-training on our data. For example, there is more than 5 points difference in all three metrics between the baseline TEC model (TEC: the second row in the table) and the model pre-trained on our pseudo pre-training (TEC+pseudo: the fourth row in the table). Our datasets are valuable not only for TEC but also for GEC. When we compare the baseline GEC model (the first row) with GEC+pseudo (the fifth row), there is about 15 points difference in BEA-2019-dev.

Comparison with previous work Kaneko et al. (2020) pseudo results (the thirteenth row) have higher scores than our GEC+LM+pseudo results (the seventh row), probably because of the difference in the size of the pseudo dataset. Theirs is about 70B, and ours is about 7B.

6 Analysis

6.1 Evaluation of TEC

In our experiment, TEC and GEC scores are not quite different in our baseline setting (the first and second row in Table 4), but TEC is consistently strong when we use a pre-trained model or pseudo data (the third to eighth rows in the figure).

These results may indicate that, in the baseline setting, Japanese information becomes noise for the TEC model, while using the pseudo-grammatical error TEC data and the pre-trained model allowed for the effective use of Japanese information. This could suggest that the *volume or quality* of the pseudo native TEC datasets may be inadequate and that these datasets alone may not allow for adequate learning of Japanese grammar and vocabulary, potentially causing the Japanese information to become noise for the TEC model. On the other hand, the models using pseudo grammatical TEC dataset or the pre-trained model are believed to learn from human-curated datasets and utilize a larger volume of data, offering both qualitative and quantitative superiority. Pseudo grammatical TEC dataset or the pre-trained model may

Error types	$F_{0.5}$ diff	Count
NOUN:INFL	15.94	10
ADJ:FORM	10.64	16
VERB:TENSE	8.66	473
SPELL	8.42	387
PUNCT	6.85	1478
VERB	5.30	402
ADV	4.69	115
ADJ	4.62	113
VERB:SVA	3.19	148
PRON	2.67	178
NOUN	2.38	328
CONJ	1.78	44
VERB:INFL	0.99	5
NOUN:NUM	0.98	251
OTHER	0.49	980
MORPH	0.05	158
NOUN:POSS	-0.37	66
PREP	-0.37	740
CONTR	-0.64	30
VERB:FORM	-0.70	236
DET	-0.87	796
WO	-2.68	95
ORTH	-3.70	352
PART	-5.01	60

Table 5: The difference in error type. The '$F_{0.5}$diff' is determined by subtracting the GEC score from the TEC score, and 'Count' represents the sum of True Positives (TP) and False Negatives (FN).

have allowed the TEC model to learn Japanese grammar and vocabulary effectively and to utilize the Japanese information more advantageously.

6.2 Error types

We compare TEC with GEC in each error type, by using BEA-2019-dev and ERRANT (Felice et al., 2016; Bryant et al., 2017). ERRANT can automatically assign errors to 25 main error categories. For instance, NOUN:INFL refers to noun inflection errors, such as "informations" corrected to "information", and VERB:TENSE represents tense-related verb errors, like "eats" corrected to "ate". In Table 5, we show the result on each error type. We use the baseline models (the first and second row in Table 4) for comparison. From these results, it appears that errors related to verbs, such as tense mistakes, might be more readily corrected with semantic information like native language sentences. On the other hand, errors involving determiner (DET: the → a), word order (WO: only can → can only), and particle (PART: in → at) may not significantly benefit from semantic information, suggesting that it may not always be useful for correcting these types of mistakes. Particularly, concepts such as determiner do not exist in Japanese, and word order sometimes differs from Japanese one. Thus, during the correction process, the information from the native language may have potentially acted as noise.

7 Conclusion

In this paper, we evaluated the grammatical error correction capabilities of TEC models and proposed a method for constructing large-scale pseudo TEC data. As a result, it was found that TEC outperforms GEC regarding grammatical error correction capabilities under many experimental settings and that our large-scale pseudo TEC dataset consistently works effectively.

Future work There are two important future works for TEC. Firstly, constructing the gold dataset of TEC is needed. Our research is evaluated on pseudo native language TEC data. It has similar grammatical errors to gold data, but there might be significant differences in native language sentences. Secondly, we hope the TEC models' ability to correct errors beyond grammar, such as semantic errors. Our experiments are limited to investigating the ability to correct grammatical errors. Unlike GEC, TEC can potentially correct semantic errors and become more educational. For example, GEC cannot correct a wrong tense because it is correct as a sentence, but TEC can.

Research on TEC is still limited, and we hope that more research will be done in the future.

Acknowledgements

We thank all the anonymous reviewers for their constructive comments. Also, I would like to express my gratitude to lab members, especially Romain Harang, Ryo Ueda, Taiga Ishii, Junjie Chen and Daichi Kato for valuable discussions.

References

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.

Guolin Cao, Takamura Hiroya, and Okumura Manabu. 2018. Multi-source neural grammatical error correction. *Proceedings of the Annual Conference of JSAI*, JSAI2018:4Pin123–4Pin123.

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.

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

Tao Ge, Furu Wei, and Ming Zhou. 2018. Fluency boost learning and inference for neural grammatical error correction. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1055–1065, Melbourne, Australia. 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, London & New York. Addison Wesley Longman.

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

Satoru Katsumata and Mamoru Komachi. 2020. Stronger baselines for grammatical error correction using a pretrained encoder-decoder model. In *Proceedings of the 1st Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing*, pages 827–832, Suzhou, China. Association for Computational Linguistics.

Seiya Kikuchi, Taisuke Onaka, Hiroaki Funayama, Yuichiroh Matsubayashi, and Inui Kentaro. 2021. 項目採点技術に基づいた和文英訳答案の自動採点 (koumokusaitengijutsunimotoduita wabuneiyakutouanno jidousaitenn:automatic scoring of japanese to english translation answers based on item scoring technology). **言語処理学会**, 第27回年次大会:690–695.

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.

Aomi Koyama, Kengo Hotate, Masahiro Kaneko, and Mamoru Komachi. 2021. Comparison of grammatical error correction using back-translation models. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Student Research Workshop*, pages 126–135, Online. Association for Computational Linguistics.

Taku Kudo and John Richardson. 2018. SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 66–71, Brussels, Belgium. Association for Computational Linguistics.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre–training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

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

Jessy Lin, Geza Kovacs, Aditya Shastry, Joern Wuebker, and John DeNero. 2022. Automatic correction of human translations. In *Proceedings of the*

2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 494–507, Seattle, United States. Association for Computational Linguistics.

Yinhan Liu, Jiatao Gu, Naman Goyal, Xian Li, Sergey Edunov, Marjan Ghazvininejad, Mike Lewis, and Luke Zettlemoyer. 2020. Multilingual denoising pre-training for neural machine translation. *Transactions of the Association for Computational Linguistics*, 8:726–742.

Masato Mita, Tomoya Mizumoto, Masahiro Kaneko, Ryo Nagata, and Kentaro Inui. 2019. Cross-corpora evaluation and analysis of grammatical error correction models — is single-corpus evaluation enough? 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 1309–1314, Minneapolis, Minnesota. Association for Computational Linguistics.

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

Makoto Morishita, Jun Suzuki, and Masaaki Nagata. 2020. JParaCrawl: A large scale web-based English-Japanese parallel corpus. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 3603–3609, Marseille, France. European Language Resources Association.

Courtney Napoles, Keisuke Sakaguchi, Matt Post, and Joel Tetreault. 2015. Ground truth for grammatical error correction metrics. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 588–593, Beijing, China. Association for Computational Linguistics.

Courtney Napoles, Keisuke Sakaguchi, and Joel Tetreault. 2017. JFLEG: A fluency corpus and benchmark for grammatical error correction. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 229–234, Valencia, Spain. Association for Computational Linguistics.

Hwee Tou Ng, Siew Mei Wu, Ted Briscoe, Christian Hadiwinoto, Raymond Hendy Susanto, and Christopher Bryant. 2014. The CoNLL-2014 shared task on grammatical error correction. In *Proceedings of the Eighteenth Conference on Computational Natural Language Learning: Shared Task*, pages 1–14, Baltimore, Maryland. Association for Computational Linguistics.

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.

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.

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.

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.