TMU Transformer System Using BERT for Re-ranking at BEA 2019 Grammatical Error Correction on Restricted Track

Masahiro Kaneko **Kengo Hotate** **Satoru Katsumata** **Mamoru Komachi**
Tokyo Metropolitan University, Japan
{kaneko-masahiro@ed, hotate-kengo@ed,
katsumata-satoru@ed, komachi@}.tmu.ac.jp

Abstract

We introduce our system that is submitted to the restricted track of the BEA 2019 shared task on grammatical error correction[1] (GEC). It is essential to select an appropriate hypothesis sentence from the candidates list generated by the GEC model. A re-ranker can evaluate the naturalness of a corrected sentence using language models trained on large corpora. On the other hand, these language models and language representations do not explicitly take into account the grammatical errors written by learners. Thus, it is not straightforward to utilize language representations trained from a large corpus, such as Bidirectional Encoder Representations from Transformers (BERT), in a form suitable for the learner's grammatical errors. Therefore, we propose to fine-tune BERT on learner corpora with grammatical errors for re-ranking. The experimental results of the W&I+LOCNESS development dataset demonstrate that re-ranking using BERT can effectively improve the correction performance.

1 Introduction

Grammatical error correction (GEC) systems may be used for language learning to detect and correct grammatical errors in text written by language learners. GEC has grown in importance over the past few years due to the increasing need for people to learn new languages. GEC has been addressed in the Helping Our Own (HOO) (Dale and Kilgarriff, 2011; Dale et al., 2012) and Conference on Natural Language Learning (CoNLL) (Ng et al., 2013, 2014) shared tasks between 2011 and 2014.

Recent research has demonstrated the effectiveness of the neural machine translation model for GEC. There are three main types of neural network models for GEC, namely, recurrent neural networks (Ge et al., 2018), a multi-layer convolutional model based on convolutional neural networks (Chollampatt and Ng, 2018a) and a transformer model based on self-attention (Junczys-Dowmunt et al., 2018). We follow the best practices to develop our system based on the transformer model, which has achieved better performance for GEC (Zhao et al., 2019).

Re-ranking using a language model trained on large-scale corpora contributes to the improved hypotheses of the GEC model (Chollampatt and Ng, 2018a). Typically, a language model is trained by maximizing the log-likelihood of a sentence. Hence, such models observe only the positive examples of a raw corpus. However, these models may not be sufficient to take into account the grammatical errors written by language learners. Therefore, we fine-tune these models trained from large-scale raw data on learner corpora to explicitly take into account grammatical errors to re-rank the hypotheses for the GEC tasks.

Bidirectional Encoder Representations from Transformer (BERT) (Devlin et al., 2019) can consider information of large-scale raw corpora and task specific information by fine-tuning on the target task corpora. Moreover, BERT is known to be effective in the distinction of grammatical sentences from ungrammatical sentences (Kaneko and Komachi, 2019). They proposed a grammatical error detection (GED) model based on BERT that achieved state-of-the-art results in word-level GED tasks. Therefore, we use BERT, pre-trained with large-scale raw corpora, and fine-tune it with learner corpora for re-ranking the hypotheses of our GEC model to utilize not only the large-scale raw corpora but also a set of information on grammatical errors.

The main contribution of this study is that

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

Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications, pages 207–212
Florence, Italy, August 2, 2019.

the experimental results demonstrate that BERT, which considers both the representations trained on large-scale and learners corpora, is effective for re-ranking the hypotheses for GEC tasks. Additionally, we demonstrated that BERT based on self-attention can re-rank sentences corrected from the GEC model by capturing long distance information.

2 TMU System

Our system is a GEC model that is combined with a re-ranker. The GEC model is given a source sentence as input and generates hypothesis sentences. These hypothesis sentences are given as input to the re-ranker, which selects the final corrected sentence form the hypothesis sentences.

We use the transformer (Vaswani et al., 2017) architecture for the GEC model because it is a state-of-the-art model in the GEC task (Zhao et al., 2019). The transformer architecture comprises multiple layers of `transformer_block`. The layers of the encoder and decoder have position-wise feedforward layers over the tokens of input sentences. The decoder has an extra attention layer over the encoder's hidden states. This GEC model is optimized by minimizing the label smoothed cross-entropy loss.

The re-ranker uses five features. We use BERT fine-tuned on learner corpora to predict the grammatical quality as a feature of re-ranking.

2.1 Architecture and training of BERT for re-ranking

We used BERT (Devlin et al., 2019) as a feature for re-ranking the hypotheses of the GEC system. BERT is designed to learn deep bidirectional representations by jointly conditioning both the left and right contexts in all layers, based on `transformer_block` with multi-head self-attention and fully connected layers. The parameters of BERT were pre-trained using a masked language model and the prediction of the next sentence.

We fine-tuned the pre-trained BERT on learner corpora to judge the grammatical quality of the input sentence, i.e., to distinguish between a sentence with and without grammatical errors on a sentence-level. We annotated sentences from parallel learner corpora having incorrect and correct sentences with 0 (incorrect) and 1 (correct) labels. Hence, using the above, we can take advantage of both the large-scale raw data and learner corpora by using BERT. The model was optimized during fine-tuning by minimizing the sentence-level cross-entropy loss.

Corpus	Train	Dev	Test
FCE	28,350	2,191	2,695
Lang-8	1,037,561	-	-
NUCLE	57,151	-	-
W&I+LOCNESS	34,308	4,384	4,477

Table 1: Number of sentences in corpora on GEC shared task for restricted track.

2.2 Re-ranking

We used the following set of features for re-ranking, which are the same as those in a previously reported approach (Chollampatt and Ng, 2018a), except for BERT:

- **GEC model**: The score of the hypothesis sentence from the GEC model is computed using the log probabilities of predictions normalized by sentence length on a token-level.
- **Language model**: A 5-gram language model score is computed by normalizing the log probabilities of the hypothesis sentence by sentence length.
- **BERT**: The predicted score for the grammatical quality of the hypothesis sentence.
- **Edit operations**: Three token level features, namely, denoting the number of substitutions, deletions, and insertions between the source sentence and the hypothesis sentence.
- **Hypothesis sentence length**: The number of words in the hypothesis sentence to penalize short hypothesis sentences.

Feature weights are optimized by minimum error rate training (MERT) (Och, 2003) on the development set.

3 Experiments

3.1 Dataset

In the restricted track, we only used the corpora listed in Table 1. The First Certificate in English (FCE) corpus (Yannakoudakis et al., 2011), Lang-8 learner corpus (Mizumoto et al., 2011), National University of Singapore Corpus of Learner

Parameter	Value
Word embedding size	500
Multi-head number	10
Layer size	6
Hidden size	2,048
Optimizer	Adam
Adam β_1	0.9
Adam β_2	0.98
Learning rate	0.0005
Learning rate scheduler	inverse square root
Warmup steps	4,000
Minimum learning rate	1e-09
Dropout	0.3
Weight decay	0.0001
Label smoothing	0.1
Max token size	4,096
Ensemble size	3

Table 2: Hyperparameter values of our transformer GEC model.

#	Team Name	P	R	$F_{0.5}$
1	UEDIN-MS	72.28	60.12	**69.47**
2	Kakao&Brain	**75.19**	51.91	69.00
7	ML@IITB	65.70	**61.12**	64.73
14	TMU	53.91	51.65	53.45

Table 3: Results of GEC systems with the highest P, R and $F_{0.5}$ overall vs TMU on restricted track on official W&I test data.

English (NUCLE) (Dahlmeier et al., 2013) and Write & Improve (W&I)+LOCNESS corpus (Yannakoudakis et al., 2018; Granger, 1998) were used for this shared task. W&I+LOCNESS corpus was a new corpus released for this shared task and the shared task systems were evaluated on a gold test set of the overall W&I+LOCNESS dataset.

We used FCE (official split of train, dev, and test set), Lang-8, NUCLE, and W&I+LOCNESS training set as training data and we split the W&I+LOCNESS development set into development and test data by random selection from each Common European Framework of Reference for Languages (CEFR) levels (beginner, intermediate, advanced, native) for the transformer and BERT. The development and test data sizes were 2,191 and 2,193, respectively.

Model	P	R	$F_{0.5}$
TMU system	37.79	**28.08**	**35.35**
w/o BERT	**38.75**	23.76	34.41
w/o language model	37.85	26.41	34.83
w/o re-ranking	36.46	22.91	32.60

Table 4: Effectiveness of re-ranking without different features.

3.2 Setup

We implemented the transformer model based on the *Fairseq* tool[2]. The hyperparameters used in our transformer GEC model are listed in Table 2. The parameters of the ensemble models were initialized with different values. We initialized the embedding layers of the encoder and decoder with the embeddings pre-trained on the English Wikipedia using *fastText* tool[3] (Bojanowski et al., 2017).

We used a publicly available pre-trained BERT model[4], namely the $\text{BERT}_{\text{BASE}}$ uncased model, which was pre-trained on large-scale BooksCorpus and English Wikipedia corpora. This model had 12 layers, 768 hidden sizes, and 16 heads of self-attention. Our model's hyperparameters for re-ranking were similar to the default ones described by Devlin et al. (2019). We used the same learner corpora with incorrect and correct sentences used for training our GEC model to fine-tune BERT.

The 5-gram language model for re-ranking was trained on a subset of the Common Crawl corpus (Chollampatt and Ng, 2018a).[5] We used a Python spell checker tool[6] on the GEC model hypothesis sentences.

3.3 Evaluation

The systems submitted to the shared task were evaluated using the ERRANT[7] scorer (Felice et al., 2016; Bryant et al., 2017). This metric is an improved version of the MaxMatch scorer (Dahlmeier and Ng, 2012) originally used in the

[2] https://github.com/pytorch/fairseq
[3] https://github.com/facebookresearch/fastText
[4] https://github.com/google-research/bert
[5] https://github.com/nusnlp/mlconvgec2018
[6] https://pypi.org/project/pyspellchecker/
[7] https://github.com/chrisjbryant/errant

(a)	**Source**	The range of public services will be expanded to remote areas , **it become** much more convenient .
	Gold	The range of public services will be expanded to remote areas , and it will become much more convenient .
	w/o BERT	The range of public services will be expanded to remote areas , *has become* much more convenient .
	TMU system	The range of public services will be expanded to remote areas , *and it will become* much more convenient .
(b)	**Source**	Her sister is 6 years old and you should **look after every weekend** .
	Gold	Her sister is 6 years old and you would have to look after her every weekend .
	w/o BERT	Her sister is 6 years old and you should *look after it every weekend* .
	TMU system	Her sister is 6 years old and you should *look after it every weekend* .

Table 5: (a) Successful and (b) unsuccessful examples of TMU system for long distance errors. **Bold** indicates the erroneous part of the source sentence; Underline indicates the corrected part of the gold sentence; *Italic* represents the corrected output of the GEC system.

CoNLL shared tasks (Ng et al., 2013, 2014). The scorer reported the performance in terms of span-based and token-based detection. The system performance was primarily measured with regard to span-based correction using the $F_{0.5}$ metric, which assigned twice as much weight to the precision. In this study, we report on precision, recall, and $F_{0.5}$ based on the ERRANT score.

3.4 Results

Table 3 presents the results of our system (TMU) and others on precision (P), recall (R) and $F_{0.5}$ on W&I+LOCNESS test data for the BEA 2019 GEC shared task on the restricted track. Our system was ranked 14 out of 21 teams.

4 Discussions

We investigated whether using BERT as a feature for re-ranking can improve the corrected results. Table 4 presents the experimental results of removing the following re-ranking features: BERT (w/o BERT); language model (w/o language model); and all features (w/o re-ranking). The recall and $F_{0.5}$ of the complete model (TMU system) is higher than those of w/o BERT, indicating that using BERT for re-ranking can improve the accuracy; especially, the recall is significantly improved. We conclude that BERT uses the advantage of large-scale raw data to acquire general linguistic expressions and learn grammatical error information from learner corpora, thus detecting and re-ranking errors more effectively.

Additionally, we analyzed the type of grammatical errors that were corrected by using BERT for re-ranking. Table 5 presents the output examples of our system with and without BERT. Example (a) demonstrates that our system can correct long distance verb tense errors, matching Gold in this case, where after stating that *"... services will be expanded ..."* in the first half, our system properly corrected *"... it become ..."* to *"... it will become ..."* in the second part of the given sentence. On the other hand, w/o BERT created a sentence with inconsistent verb tense by changing *"... it become ..."* to *"... it has become ..."*. Example (b) demonstrates that neither of the systems, i.e., with and without BERT, could properly correct the coreference resolution error. They both failed to trace the reference of *"it"* to *"her sister"*. By using BERT based on self-attention for re-ranking, which is effective for long distance information, our system became better at solving long distance errors; however, there is a room for improvement.

5 Related Work

Re-ranking using a language model trained on large-scale raw data significantly improved the results in numerous GEC studies (Junczys-Dowmunt and Grundkiewicz, 2016; Chollampatt and Ng, 2018a; Grundkiewicz and Junczys-Dowmunt, 2018; Junczys-Dowmunt et al., 2018; Zhao et al., 2019). However, their models do not explicitly consider grammatical errors of language learners.

Yannakoudakis et al. (2017) utilized the score from a GED model as a feature to consider grammatical errors for re-ranking. Chollampatt and Ng (2018b) proposed a neural quality estimator for GEC. Their models predict the quality score when given a source sentence and its corresponding hypothesis. They consider representations of grammatical errors of learners for re-ranking. However, their models did not use large-scale raw corpora.

Rei and Søgaard (2018) used a sentence-level GED model based on bidirectional long short-term memory (LSTM). The goal of their study was to predict the token-level labels on a sentence-level using the attention mechanism for zero-shot sequence labeling.

Kaneko and Komachi (2019) proposed a model of applying attention to each layer of BERT for GED and achieved state-of-the-art results in word-level GED tasks. Our BERT model predicts grammatical quality on a sentence-level for re-ranking.

6 Conclusion

In this paper, we described our TMU system, which is based on the GEC transformer model using BERT for re-ranking. We evaluated our TMU system on the restricted track of the BEA 2019 GEC shared task. The experimental results demonstrated that using BERT for re-ranking can improve the correction quality.

In this work, we only considered the information of the hypothesis sentence. In our future work, we will analyze the re-ranker, allowing BERT to utilize the information of the source sentence of the GEC model as well, given both source and hypothesis sentences.

7 Acknowledgments

We thank Yangyang Xi of Lang-8, Inc. for kindly allowing us to use the Lang-8 learner corpus. This work was partially supported by JSPS Grant-in-Aid for Scientific Research (C) Grant Number JP19K12099.

References

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching Word Vectors with Subword Information. *TACL*, 5:135–146.

Christopher Bryant, Mariano Felice, and Ted Briscoe. 2017. Automatic Annotation and Evaluation of Error Types for Grammatical Error Correction. In *ACL*, pages 793–805, Vancouver, Canada. Association for Computational Linguistics.

Shamil Chollampatt and Hwee Tou Ng. 2018a. A Multilayer Convolutional Encoder-Decoder Neural Network for Grammatical Error Correction. In *AAAI*, pages 5755–5762, New Orleans, Louisiana. Association for the Advancement of Artificial Intelligence.

Shamil Chollampatt and Hwee Tou Ng. 2018b. Neural Quality Estimation of Grammatical Error Correction. In *EMNLP*, pages 2528–2539, Brussels, Belgium. Association for Computational Linguistics.

Daniel Dahlmeier and Hwee Tou Ng. 2012. Better Evaluation for Grammatical Error Correction. In *NAACL*, pages 568–572, Montreal, 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 *BEA*, pages 22–31, Atlanta, Georgia. Association for Computational Linguistics.

Robert Dale, Ilya Anisimoff, and George Narroway. 2012. HOO 2012: A Report on the Preposition and Determiner Error Correction Shared Task. In *BEA*, pages 54–62, Montreal, Canada. Association for Computational Linguistics.

Robert Dale and Adam Kilgarriff. 2011. Helping Our Own: The HOO 2011 Pilot Shared Task. In *ENLG*, pages 242–249, Nancy, France. 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 *NAACL*, Minneapolis, USA. 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 *COLING*, 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 *ACL*, 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. pages 3–18. Learner English on Computer.

Roman Grundkiewicz and Marcin Junczys-Dowmunt. 2018. Near Human-Level Performance in Grammatical Error Correction with Hybrid Machine Translation. In *NAACL*, pages 284–290, New Orleans, Louisiana. Association for Computational Linguistics.

Marcin Junczys-Dowmunt and Roman Grundkiewicz. 2016. Phrase-based Machine Translation is State-of-the-Art for Automatic Grammatical Error Correction. In *EMNLP*, pages 1546–1556, Austin, Texas. 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 *NAACL*, pages 595–606, New Orleans, Louisiana. Association for Computational Linguistics.

Masahiro Kaneko and Mamoru Komachi. 2019. Multi-Head Multi-Layer Attention to Deep Language Representations for Grammatical Error Detection. In *CICLing*, La Rochelle, France.

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 *IJCNLP*, pages 147–155, Chiang Mai, Thailand. Asian Federation of Natural Language Processing.

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 *CoNLL*, 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 *CoNLL*, pages 1–12, Sofia, Bulgaria. Association for Computational Linguistics.

Franz Josef Och. 2003. Minimum Error Rate Training in Statistical Machine Translation. In *ACL*, pages 160–167, Sapporo, Japan. Association for Computational Linguistics.

Marek Rei and Anders Søgaard. 2018. Zero-Shot Sequence Labeling: Transferring Knowledge from Sentences to Tokens. In *NAACL*, pages 293–302, New Orleans, Louisiana. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is All you Need. In *NIPS*, pages 5998–6008. Curran Associates, Inc.

Helen Yannakoudakis, Øistein E. Andersen, Ardeshir Geranpayeh, Ted Briscoe, and Diane Nicholls. 2018. Developing an Automated Writing Placement System for ESL Learners. pages 251–267. Applied Measurement in Education 31:3.

Helen Yannakoudakis, Ted Briscoe, and Ben Medlock. 2011. A New Dataset and Method for Automatically Grading ESOL Texts. In *NAACL*, pages 180–189, Portland, Oregon, USA. Association for Computational Linguistics.

Helen Yannakoudakis, Marek Rei, Øistein E. Andersen, and Zheng Yuan. 2017. Neural Sequence-Labelling Models for Grammatical Error Correction. In *EMNLP*, pages 2795–2806, Copenhagen, Denmark. 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 *NAACL*, Minneapolis, USA. Association for Computational Linguistics.