QBS
Vol. 39, No. 1, pp. 17-24 (2020)
https://doi.org/10.22283/qbs.2020.39.1.17
pISSN 2288-1344 eISSN 2508-7185 **Original Article**

Quantitative Bio-Science
Official Journal of the Institute of Natural Science at Keimyung University
Available Online at https://qbs.kmu.ac.kr:442/

Grammatical Error Correction Models for Korean Language via Pre-trained Denoising

Jin Hong Min[1], Seong Jun Jung[1], Se Hee Jung[2], Seongmin Yang[1], Jun Sang Cho[1], Sung Hwan Kim[1,*]

[1]Department of Applied Statistics, Konkuk University, Korea
[2]AI Analytics Team, DeepVisions, Seoul, Korea
(Received April 9, 2020; Revised May 7, 2020; Accepted May 12, 2020)

ABSTRACT

Since the era of burgeoning big data, text data have not only become increasingly accessible but also been widely applied to diverse domains. In these circumstances, adequate language processing is urgently required to handle the enormous amount of unorganized data (*e.g.*, wrong, missing, incomplete). To deal with text data errors, varied efforts have been applied to develop grammatical error correction (GEC) models, especially for the English language. However, correction models for Korean have remained relatively unexplored. In this paper, we propose a novel GEC model specialized in Korean. Owing to the lack of training sample-label pairs (parallel corpus) in the pre-training phase, prior to training in the main stage, this model accommodates a pre-defined noise function that produces artificial errors to reinforce the previous language-correction models. For numerical study, we generate approximately 37 million training sentences and choose the case study of Korean learners' parallel corpus, a benchmark dataset for text correction. We conclude from the study that the proposed model outperforms humans in the context of bilingual evaluation understudy scores.

Key words : Natural language processing, Grammatical error correction, Denoising auto-encoder, Transfer learning, Parallel corpus

1. Introduction

GEC task is one of natural language process (NLP) tasks modifying errors in sentences, aiming to make sentences straightforward and easy-to-read. The GEC model deals with not just correcting errors (*e.g.*, typos) but also adjusting mood of sentence. The following Korean sentence, "이 지역은 무단 입산자에 대하여는 자연 공원법 제60조에 의거 처벌을 받게 됩니다" (→ "i jiyeog-eun mudan ibsanja-e daehayeoneun jayeon gong-wonbeob je60jo-e uigeo cheobeol-eul badge doebnida") is should be replaced with "이 지역은 자연 공원법 제60조에 의거하여 무단 입산자를 처벌하는 곳입니다" (→ "i jiyeog-eun jayeon gong-wonbeob je60jo-e uigeohayeo mudan ibsanjaleul cheobeolhaneun gos-ibnida") in view of both natural word order and subject-verb relationship. The quality of GEC highly depends on the degree of understanding semantic context of sentences. Especially it is important to grasp longer dependencies (*e.g.*, non-idiomatic expressions or contextual errors) allowing for the relationship between a subject and a verb. However, existing methods are limited in scope to modifying elementary grammatical errors at most, relying on several statistical techniques [1]. In general, the GEC model learns correction information from a paral-

* Correspondence should be addressed to Sung Hwan Kim, Assistant Professor, Department of Applied Statistics, Konkuk University, Seoul 05029, Korea. Tel: Tel: +82-2-450-3658, E-mail: shkim1213@konkuk.ac.kr
The first two authors should be regarded as joint first authors. JH Min and SJ Jung are undergraduate students, Department of Applied Statistics, Konkuk University, Seoul 05029, Korea. SH Jung is a research scientist, AI Analytics Team, DeepVisions, Seoul 03752, Korea. S Yang is a Ph.D student and JS Cho is a post-doc fellow, Department of Applied Statistics, Konkuk University, Seoul 05029, Korea.

lel corpus consisting of both original sentences and corrected sentences. Different from a neural network-based transition, the GEC hardly fits adequately due to absence of sufficient data, making it difficult to achieve satisfactory results [2]. As the promising solution to data scarcity, the transfer learning has garnered an increasing attention in the blessing of its useful knock-on effects. Recent works [3-5] prove to be a valid way to enhance network performance. It is especially notable that Xie et al. [6] applied a concept of denoising auto-encoder (DAE) primarily used for image processing. To this end, they artificially added synthetic errors (noises) to original sentences. Using these erroneous data, they trained the network in advance and improved the existing GEC model [6]. Inspired by this study, we propose a scheme for pre-learning of the DAE with the noise function reflecting the characteristics of Korean language. In the stage of training, our model intensively corrects realistic Korean errors generated by the proposed noise. This greatly helps understand actual usage patterns of Korean. Needless to say, the appropriate level of complexity in training and is a key factor to determine network performance. In this sense, the way to generate noises in our experiment is largely two-fold: a syntactic word-based method and a morpheme-based method, from which large scale datasets for pre-training have been created to pre-train GEC models. In this process, approximately 37 million sentences including Ko-Wiki (https://ko.wikipedia.org/), Sejong Corpus (https://ithub.korean.go.kr/user/guide/corpus/guide1.do), and Naver Terms (https://terms.naver.com/) were used. In the virtue of the great volume of high-quality datasets, our model attained a high level of literacy better than human.

The rest of this paper is as follows. In Section 2, we introduce related works. the approach accommodates the characteristics of Korean on GEC in Section 3. Section 4 provides the experimental setups. We discuss results in Section 5.

2. Related Work

2.1 Denoising autoencoder

Denoising Auto-Encoder (DAE) [7] is a type of unsupervised learning for reconstruction of data with artificial noise added. Specifically, it operates through encoder recognizing fundamental, stable representations of partially destructed inputs (*i.e.*, noise-added) and decoder reconstructing pure inputs from the features. For noise-added inputs, the aim of DAE is to minimize the following loss function:

$$Loss\,(\boldsymbol{x}, Decoder\,(Encoder\,(\boldsymbol{x}+\boldsymbol{noise}))).$$

As discussed in Vincent et al. [7], the fundamental background of denoising autoencoder is robustness to partial destruction of inputs. Although an input is partially destroyed with noise, resultant inputs should have almost the same representation. Because the intrinsic nature of pure input (before noise-added) itself is still the same. In the case of applying DAE to Korean sentence correction. it is essential to customize the proper noise function producing practical Korean errors. Unless this is the case, it is likely to turn out to be satisfactory results due to deterioration of training data quality. Recently Zhao et al. [8] tried several ways to generate noise tokens. They first arbitrarily deleted, added or exchanged tokens (syntactic words) with a probability of 10%, respectively. Combining the processed tokens with normal distribution bias, they distorted the pure tokens (*i.e.*, correct).

However, the absence of sophisticated noise functions is the pain point. As the noise for training of DAE, the resultant tokens are effective but still meager to secure generality in terms of data diversity. Given that the GEC model accepts the stage of pre-training, Lample et al. [9] takes a similar approach to ours. The pre-trained DAE rectifies erroneous sentences via word-by-word translations. However, since the model is built on the low-sources (*i.e.*, data scarcity), they also cannot generalize well and may produce wrong sentences. While a large number of noising techniques have been developed for digital image data (*e.g.*, Gaussian noise), text data have been relatively less focused. To address this challenge, regularization techniques have been adopted for enabling to drop or replace individual tokens. Nevertheless, none of the models takes Korean into account yet.

2.2 Transfer learning

Transfer learning is one of the learning techniques that reuse the trained models derived from data-rich environments. This has the advantage of building models even in the cases where there are not enough data for training. Since the cases are still rife in various fields, it is not confined to the GEC only, widely being used. Taking advantages of transfer learning, several outstanding models have been developed. The learned weights are transferred as the initial values for main-training. Similar to the Inception [10], the ResNet5 [11] as pre-model-based networks for image classification, we select a model well-learned and relevant to the task. The ELMo [3] proposes to extract context-sensitive features from a language

model. The OpenAI GPT [12] enhances the context-sensitive embedding by adjusting the Transformer [13]. BERT [4], however, adopts a masked language model while adding a next sentence prediction task into the pre-training.

The biggest challenge of GEC task is that the model has difficulty in learning due to the data scarcity. As the prerequisite for transfer learning, it is required to introduce pre-trained models. To this end, we adopted the word embedding method. It is a traditional method for effective learning of general language representation, training a language model in advance with a large amount of unannotated data. For instance, the two models (*e.g.*, Word2Vec [14] and GloVe [15]) are well-known to learn embedding vectors the results of the word embedding with word co-occurrence from a large corpus of text.

3. Proposed Method

Our aim is to create a noise function that reflects the characteristics of Korean and thereby produces realistic Korean errors. To this end, we introduce two ways of generating noise: a syntactic word-based approach and a morpheme-based approach.

In the syntactic word-based approach, we determine whether the given word is a predicate with the help of the Soylemma library (https://pypi.org/project/soylemma/), python library for Korean NLP. If it is a predicate, we separate and extract the stem and the ending from the word. Converting the ending, we generate the conjugated forms of the word. Then, we replace the given predicate with one of the generated conjugated forms with equal probabilities. For example, the word "사다" (→ "sada") can be converted into various conjugated forms such as "사는" (→ "saneun"), "샀다" (→ "sassda"), "사고" (→ "sago"), etc. If it is not a predicate, we modify the word in reference to the "Frequently Wrong Korean List" in Namuwiki (https://namu.wiki/; *e.g.*, "얻다 대고" (→ "eodda daego") to "어따 대고" (→ "eotta daego"), "얼마큼" (→ "eolmakeum") to "얼만큼" (→ "eolmankeum")), which deals with grammatical errors that even Koreans often mistakes, such as typos or spacing errors.

In the morpheme-based approach, we extract a variety of Korean errors from the Korean Learners' Corpus (https://kcorpus.korean.go.kr/service/goSummaryStatus.do), correctional materials for texts written by Korean learners. And we collect edits that occur at least three times to avoid overfitting this dataset. Accordingly, we obtain an auto-configuration dictionary of common edits created by human annotators in a training set. Built on these errors, a noise function is generated by replacing the given correct morphemes randomly.

Using the two ways of generating noise, we generated a noise dataset containing nearly 37 million sentences to pre-train our model. Taken together, we present in Fig. 1 the workflows the proposed method for better understanding.

4. Experiments

In the stage of modeling, we use the fairseq [16], a sequence modeling python package, that allow to customize models for translation, summarization, language modeling and other text generation tasks. The final GEC model was determined in two stages. To verify the actual usefulness of transfer learning, we first compared a model based on transfer-learning with existing models. As an evaluation metric, we exploited the Bilingual Evaluation Understudy (BLEU) [17], an algorithm for evaluating the quality of texts machine-translated from one natural language to another. The metric BLEU is motivated by n-grams. The BLEU is a method of measuring performance for translation, comparing similarity between machine translation and human translation task.

4.1 Dataset and preparing

Table 1 represents all working data sources, sizes and parallel corpus. For pre-training, we use Korean Wikipedia (Ko wiki), Naver Knowledge Encyclopedia (Naver terms), and Sejong corpus (Sejong corpus). Early experiments have shown that the quality of pre-training data is crucial to the performance of the final model, as our DAE model assumes that this unannotated corpus contains few grammatical errors.

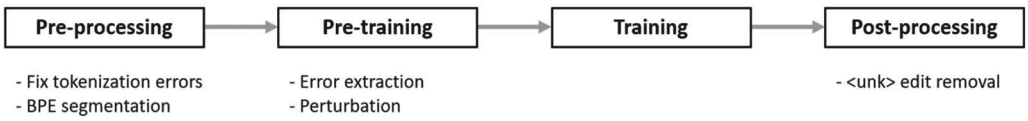

Fig. 1. The outline of the sequential transfer learning using transformers.

Table 1. The summary of the datasets. The first three datasets are used for pre-training, and the fourth data are used for training, validation, and test

	Naver terms	Ko wiki	Sejong corpus	Korean Learners' Corpus		
				Train	Validation	Test
# sentences	20,225,448 (20 M)	1,446,140 (1.4 M)	15,359,535 (15.3 M)	33,331 (33 K)	4,166 (4 K)	4,167 (4 K)
Paralleled annotation	X	X	X	O	O	O

Table 2. The summary of data information used for each stage

Step	Dataset
Error extraction	Korean Learners' Corpus
Pre-training	Ko-wiki, Naver terms, Sejong corpus
Training	Korean Learners' Corpus - training
Validation	Korean Learners' Corpus - validation
Test	Korean Learners' Corpus - test

The type of Wikipedia text is well-known for high quality, and Sejong corpus consists of modern corpus, newspapers, magazines, and books that convey specific information. In that sense, such materials have few grammatical errors. Our last pre-training data are a collection of about 37M sentence pairs (noised, corrected) based on these data sets, and our noise scenarios. To address the lack of datasets for pre-training, we make realistic noise function to generate perturbed versions of large unannotated corpora. The resulting parallel corpora are subsequently used to pre-train the transformer models as the DAE. This function captures grammatical errors within the domain that Korean learners usually mistake.

4.2 Pre-processing

In this stage, we use a regular expression to resolve minor tokenization problems. Corrected version of these fixed inputs were fed into the rule-based Korean spell checker (hanspell; https://github.com/9beach/hanspell/) and Korean spacing corrector (PyKoSpacing; https://github.com/haven-jeon/PyKoSpacing) because the target sentences have no grammatical errors for GEC tasks. Before feeding spellchecked text into our seq2seq model, we apply byte-pair encoding (BPE) [18] using SentencePiece [19]. We first train the SentencePiece model with 1.4M sentence size on the Korea Wikipedia corpus, and apply this model to all input text. This allows to avoid unknown tokens in all of our datasets, including corpus of Korean learners.

4.3 Modeling and training

In our experiments, we use two variants of the transformer model; the base Transformer [13] and the copy-augmented Transformer [8]. As discussed in Seo et al. [20] deep representation abilities of networks has a great effect on model performance. Inspired by this, we deeply stacked our networks. For the base transformer model, we use the model of six blocks ranging from 512 to 2048 units with eight attention heads and pre-attention layer normalization. For the copy-augmented Transformer, we follow the default configurations with six blocks of ranging from 512 to 4096 units with eight attention heads, along with an 8-head copy attention layer. Our model training is a two-stage process, as illustrated in Fig. 1: DAE pre-training and training. Each step will train the model until the weights of pre-learning converge, and the learned weights are transferred to the next steps. In all training steps, we used the Adam optimizer [21].

In the Fig. 2, the term, 6x, corresponds to 6 layers, each of which is made of 2 sub-layers. One is the multi-head attention mechanism, and another is the simple fully connected network. x_j is the source sentence, y_j is the target sentence and the two terms, h^{trg} and h^{src}, represents the decoder's current hidden states and the encoder's hidden states, where $j = 1,2,3\cdots$. p is the final probability considering the proportion of generation and copy mechanism, and the proportion is determined by the balancing factor, α_t^{copy}.

4.4 Results

Table 3 summarizes the BLEU scores of Korean Leaners' parallel corpus which represent how much the predicted and the true sentence overlap. It is shown that base transformer models (BLEU = 82.51) perform better than copy-augmented transformer models (BLEU = 47.48). It is remarkable in the sense that when BLEU is higher than 80, the model is believed to be superior to human. However, the copy-augmented transformer model does not significantly improve (BLEU = 47.48) compared to the original copy-augmented transformer without the

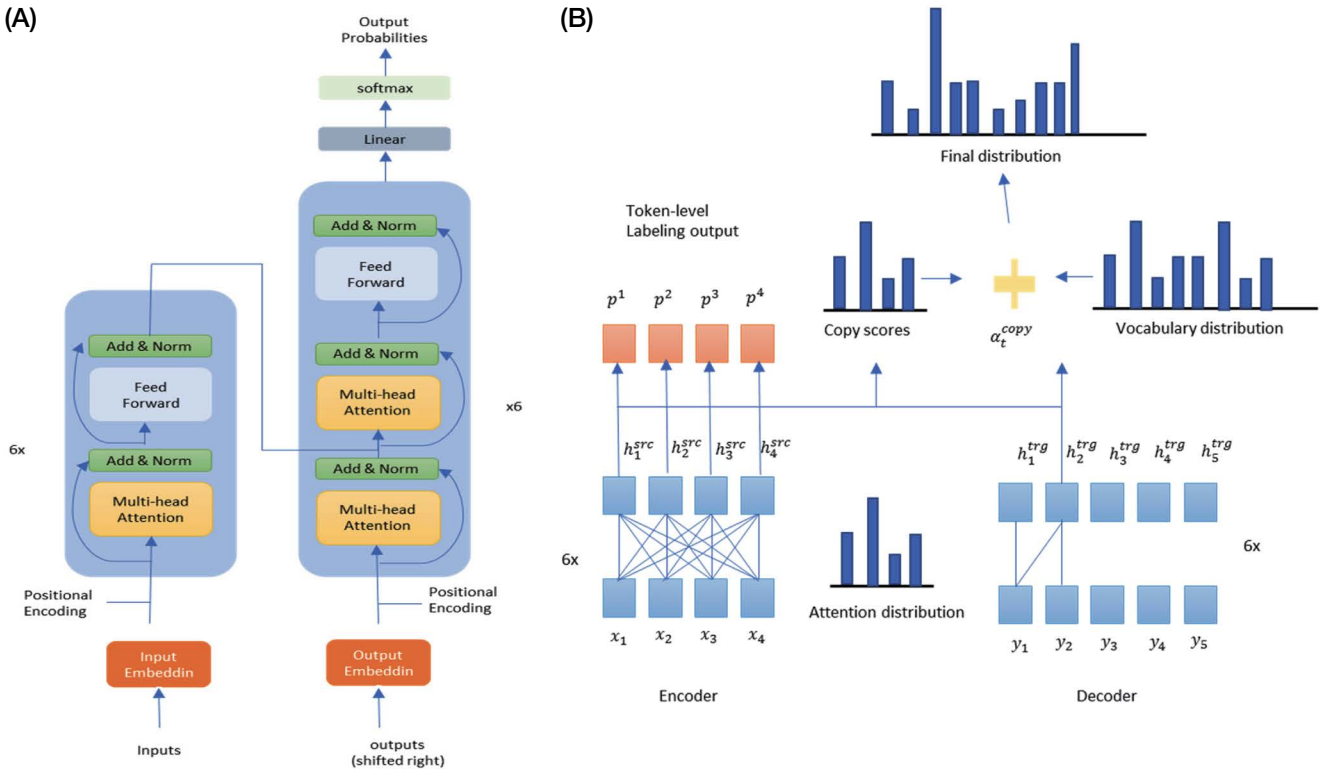

Fig. 2. (A) The transformer model architecture and (B) copy-augmented transformer model architecture.

Table 3. BLEU scores in each step for Korean Leaners' parallel corpus. Δ refers to the difference in BLEU between pre-train and train models

Steps	Validation		Test	
	BLEU	Δ	BLEU	Δ
DAE pre-train (base-transformer)	40.22	n/a	40.10	n/a
Train (base-transformer)	83.07	42.85	82.51	42.41
DAE pre-train (copy-augmented transformer)	46.10	n/a	47.09	n/a
Train (copy-augmented transformer)	47.19	1.09	47.48	0.39

DAE (BLEU = 47.09).

In what follows in Table 4, we show that grammatical correction results from base and copy-augmented transformer models. In the stage of comparison, we used a dataset, the Korean learners' parallel corpus, containing a variety of Korean word/morpheme-related errors, aiming to evaluate the learned models' understanding of Korean syntax. Given the characteristics of the dataset, its large volume and focus on grammatical fundamentals (*e.g.*, token, morpheme), the dataset could reveal appreciable difference of the models in Korean correction. Additionally, in the dataset, there are various error patterns such as one or more wrong syllables in a word, spacing errors, wrong particles, etc.

In Input sentence 1, the base model modified "컨퓨터" (→ "keonpyuteo") to "컴퓨터" (→ "keompyuteo"), but the copy-augmented model modified "컨퓨터" (→ "keonpyuteo") to "컨터터" (→ "keonteoteo"). This shows that the base model changes the wrong noun to the correct noun, but the copy-augmented model fails. In addition, in Input sentence 3, the base model corrected the incorrect noun "푹촌" (→ "pugchon") to "북촌" (→ "bugchon") and "싸람" (→ "ssalam") to "사람" (→ "salam"), but the copy-augmented model failed to correct properly. In Input sentence 2, the base model modified "것이" (→ "geos-i") to "것을" (→ "geos-eul") to correct postposition (Korean particle), but the copy-augmented model failed to identify the false grammar. In addition, in Input sentence 3, the base model the base model added the correct postposition as "한옥" (→ "han-og") as "한옥을" (→ "han-og-eul"), but the copy-augmented model failed. In addition, in Input sentence 4,

Table 4. The example of grammatical correction from the base and copy-augmented transformer models

Input sentence 1	일본에 가면 한국어를 잊어버리니까 컨퓨터로 한국 영화를 많이 볼 거에요 → "ilbon-e gamyeon hangug-eoleul ij-eobeolinikka keompyuteolo hangug yeonghwaleul manh-i bol geoyeyo"
Model correction (base-transformer)	일본에 가면 한국어를 잊어버리니까 컴퓨터로 한국 영화를 많이 볼 거에요 → "ilbon-e gamyeon hangug-eoleul ij-eobeolinikka keompyuteolo hangug yeonghwaleul manh-i bol geoeyo"
Model correction (copy-augmented transformer)	일본에 가면 한국어를 잊어버리니까 컨터터로 한국 영화를 많이 많이 볼 거에요 → "ilbon-e gamyeon hangug-eoleul ij-eobeolinikka keonteoteolo hangug yeonghwaleul manh-i manh-i bol geoeyo"
Human correction	일본에 가면 한국어를 잊어버리니까 컴퓨터로 한국영화를 많이 볼 거에요 → "ilbon-e gamyeon hangug-eoleul ij-eobeolinikka keompyuteolo hangug-yeonghwaleul manh-i bol geoeyo"
Input sentence 2	일주일에 한 번 이상 카페에 간다는 것이 알게 됐습니다 → "ilju-il-e han beon isang kapee gandaneun geos-i alge dwaessseubnida"
Model correction (base-transformer)	일주일에 한 번 이상 카페에 간다는 것을 알게 되었습니다 → "ilju-il-e han beon isang kapee gandaneun geos-eul alge doeeossseubnida"
Model correction (copy-augmented transformer)	일주일에 한 번 이상 카페에 간다는 것이 알게 알습니다 → "ilju-il-e han beon isang kapee gandaneun geos-i alge alseubnida"
Human correction	일주일에 한 번 이상 카페에 간다는 것을 알게 되었습니다 → "ilju-il-e han beon isang kapee gandaneun geos-eul alge doeeossseubnida"
Input sentence 3	그리구 대부분 한국 싸람들은 푹촌 한옥 마을과 전주에 있는 한옥 소개하고 싶어했습니다 → "geuligu daebubun hangug ssalamdeul-eun pugchon han-og ma-eulgwa jeonjue issneun han-og sogaehago sip-eohaessseubnida"
Model correction (base-transformer)	그리고 대부분 한국 사람들은 북촌 한옥마을과 전주에 있는 한옥을 소개하고 싶어 했습니다 → "geuligo daebubun hangug salamdeul-eun bugchon han-ogma-eulgwa jeonjue issneun han-og-eul sogaehago sip-eo haessseubnida"
Model correction (copy-augmented transformer)	그리구 대부분 한국 싸람들은 푹촌 한옥과 전주에 있는옥 소개하고 싶어 싶어 싶습니다 → "geuligu daebubun hangug ssalamdeul-eun pugchon han-oggwa jeonjue issneun-og sogaehago sip-eo sip-eo sipseubnida"
Human correction	그리고 대부분 한국 사람들은 북촌 한옥마을과 전주에 있는 한옥을 소개하고 싶어 했습니다 → "geuligo daebubun hangug salamdeul-eun bugchon han-ogma-eulgwa jeonjue issneun han-og-eul sogaehago sip-eo haessseubnida"
Input sentence 4	그리고 여행하기커녕 주말에 쉬는 시간도 없었다 → "geuligo yeohaenghagikeonyeong jumal-e swineun sigando eobs-eossda"
Model correction (base-transformer)	그리고 여행하기는커녕 주말에 쉬는 시간도 없었다 → "geuligo yeohaenghagineunkeonyeong jumal-e swineun sigando eobs-eossda"
Model correction (copy-augmented transformer)	그리고 여행하기커녕 주말에 쉬는 시간도 없었다 → "geuligo yeohaenghagikeonyeong jumal-e swineun sigando eobs-eossda"
Human correction	그리고 여행하기는커녕 주말에 쉴 시간도 없다 → "geuligo yeohaenghagineunkeonyeong jumal-e swil sigando eobsda"

the base model modified the sentence by adding the correct assistant postposition to "여행하기커녕" (→ "yeohaenghagikeonyeong") and "여행하기는커녕" (→ "yeohaenghagineunkeonyeong"). Evaluating the correction results with the

BLEU score, we confirmed the superior performance of our proposed model, especially in the correct use of syntactic words and functional morphemes.

5. Discussion

In this article, we propose the language correction model to utilize realistic error function that builds on Korean language features. The generated synthetic corpus by error function was used in pre-training transformer model. Taking all results together, we conclude that our proposed model produces outstanding BLEU scores compared to base-transformer. For future work, the followings are further research topics worth to focus. Typically, the Korean GEC (*e.g.*, agglutinative language) is not functional as expected, mainly attributed to its own language complexity. In this regard, there are still much room to improve practical utility. It is common that the pre-processed corpus contains inevitable errors. This eventually leads to the model mistaking the same errors. To circumvent this challenge, it is worth to refine data annotations error made by human. In addition, it is interesting to figure out the rationale behind why the base transformer model performs better the copy-augmented transformer model in analytic standpoints. It is also interesting to note that Zhou et al. [22] makes synthetic English corpus through low-quality and high-quality machine-translation to create the paralleled corpus in Chinese. The idea behind is that the model pairs both low-quality machine-translation as inputs and high-quality machine-translation as outputs. Inspired, it is worthwhile to see if this method is applicable to synthetic Korean corpus.

Acknowledgements

This research was supported National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (2020R1C1C1A01005229 and 2019R1I1A1 A01061824).

References

1. Sidorov G, Gupta A, Tozer M, Catala D, Catena A, Fuentes S. Rule-based system for automatic grammar correction using syntactic n-grams for english language learning (l2). Conf. on Computational Natural Language Learning: Shared Task: 96-101. Aug. 2013.
2. Lample G, Ott M, Conneau A, Denoyer L, Ranzato MA. Phrase-based & neural unsupervised machine translation. ArXiv Preprint ArXiv:1804.07755 2018.
3. Peters ME, Neumann M, Iyyer M, Gardner M, Clark C, Lee K, et al. Deep contextualized word representations. ArXiv Preprint ArXiv:1802.05365 2018.
4. Devlin J, Chang MW, Lee K, Toutanova K. Bert: Pre-training of deep bidirectional transformers for language understanding. ArXiv Preprint ArXiv:1810.04805 2018.
5. Yang Z, Dai Z, Yang Y, Carbonell J, Salakhutdinov RR, Le QV. Xlnet: Generalized autoregressive pretraining for language understanding. Adv Neur In 2019;5754-5764.
6. Xie Z, Genthial G, Xie S, Ng AY, Jurafsky D. Noising and denoising natural language: Diverse backtranslation for grammar correction. Conf. of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies: 1; 619-628. Jun. 2018.
7. Vincent P, Larochelle H, Bengio Y, Manzagol PA. Extracting and composing robust features with denoising autoencoders. Int'l Conf. on Machine Learning: 1096-1103. Jul. 2008.
8. Zhao W, Wang L, Shen K, Jia R, Liu J. Improving grammatical error correction via pre-training a copy-augmented architecture with unlabeled data. ArXiv Preprint ArXiv:1903.00138 2019.
9. Lample G, Ott M, Conneau A, Denoyer L, Ranzato MA. Phrase-based & neural unsupervised machine translation. ArXiv Preprint ArXiv:1804.07755 2018.
10. Szegedy C, Liu W, Jia Y, Sermanet P, Reed S, Anguelov D, et al. Going deeper with convolutions. IEEE Conf. on Computer Vision and Pattern Recognition: 1-9. 2015.
11. He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. IEEE Conf. on Computer Vision and Pattern Recognition: 770-778. 2016.
12. Radford A, Narasimhan K, Salimans T, Sutskever I. Improving language understanding by generative pre-training. https://s3-us-west-2.amazonaws.com/openai-assets/researchcovers/languageunsupervised/language understanding paper. pdf. 2018.
13. Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, et al. Attention is all you need. Adv Neur In 2017;5998-6008.
14. Mikolov T, Sutskever I, Chen K, Corrado GS, Dean J. Distributed representations of words and phrases and their compositionality. Adv Neur In 2013;3111-3119.
15. Pennington J, Socher R, Manning CD. Glove: Global vectors for word representation. Conf. on Empirical Methods in Natural Language Processing (EMNLP): 1532-1543. Oct. 2014.
16. Ott M, Edunov S, Baevski A, Fan A, Gross S, Ng N, et al. fairseq: A fast, extensible toolkit for sequence modeling. ArXiv Preprint ArXiv:1904.01038 2019.
17. Papineni K, Roukos S, Ward T, Zhu WJ. BLEU: a method for automatic evaluation of machine translation. Annual Meeting on

Association for Computational Linguistics: 311-318. Jul. 2002.
18. Sennrich R, Haddow B, Birch A. Neural machine translation of rare words with subword units. ArXiv Preprint ArXiv:1508.07909 2015.
19. Kudo T, Richardson J. Sentencepiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. ArXiv Preprint ArXiv:1808.06226 2018.
20. Yunbeom S, Changha H. Predicting bitcoin market trend with deep learning models. QBS 2018;37:65-71.
21. Kingma DP, Ba J. Adam: A method for stochastic optimization. ArXiv Preprint ArXiv:1412.6980 2014.
22. Zhou W, Ge T, Mu C, Xu K, Wei F, Zhou M. Improving Grammatical Error Correction with Machine Translation Pairs. ArXiv Preprint ArXiv:1911.02825 2019.