Korean Grammatical Error Correction based on Transformer with Copying Mechanisms and Grammatical Noise Implantation Methods*

Myunghoon Lee
Kyonggi University
Republic of Korea
vhxmpqms@naver.com

Hyeonho Shin
Kyonggi University
Republic of Korea
shinhh9554@gmail.com

Dabin Lee
Kyonggi University
Republic of Korea
leedabin0616@gmail.com

Sung-Pil Choi
Kyonggi University
Republic of Korea
sungpil@gmail.com

ABSTRACT
Grammatical Error Correction (GEC) is the task of detecting and correcting various grammatical errors in texts. Many previous approaches to the GEC have used various mechanisms including rules, statistics and their combinations. Recently, the performance of the GEC in English has been drastically enhanced due to the vigorous applications of deep neural networks and pretrained language models. Following the promising results of the English GEC tasks, we apply the Transformer with the copying mechanism into the Korean GEC task by introducing novel and effective noising methods for constructing Korean GEC datasets. Our comparative experiments showed that the proposed system outperforms two commercial grammar check and correction services in various aspects.

KEYWORDS
Grammatical Error Correction (GEC), Neural Machine Translation (NMT), Transformer, Copying Mechanism

1 INTRODUCTION
Grammatical Error Correction (GEC) is the task of automatically detecting and correcting various types of grammatical errors and typos in texts. It typically focuses on all the textual mistakes and errors including morphological, lexical, syntactic and semantic irregularities that could be appeared in texts [1].

Until now, almost all the previous approaches to GEC for Korean have utilized the rule-based methods where all the target error patterns as well as corresponding correction logics should be recognized in advance and consistently expanded [2]. However, it is obvious that the rule-based mechanisms have a disadvantage in that they require much of manual labor in achieving the error patterns and correction logics. Furthermore, it is unlikely to promptly reflect a radical change in the current linguistic environment such as the rise of newly coined words and the natural extinction of old-fashioned words and syntactic rules [1].

To address the limitations and problems mentioned earlier, many researchers are now attempting to apply Neural Machine Translation (NMT) models for the GEC because they are perfectly appropriate for the task translating grammatically incorrect sentences to correct sentences. The NMT-based models have two advantages. Firstly, their neural encoder-decoder mechanism effectively encodes various grammatical errors in training data and generates the corresponding corrected texts based on the encoded information [3]. In addition, their error handling coverage is much broader than the conventional methods even handling infrequent and rare error patterns with the generalization ability of the mechanism [1]. These strength of the models leads to the remarkable performance improvements in the recent English GEC tasks showing the promising potentials of the approaches as a future research direction [3].

In this paper, we introduce an effective Korean Grammatical Error Correction model based on Transformer equipped with a copying mechanism and various noising methods for automatically generating a training set. It is shown that during the GEC execution, about 80% of input texts remain unchanged and only 20% are recognized as errors and thus the system changes their lexical and syntactic structures. The copying mechanisms can effectively cope with the phenomenon by enhancing the preservation capability of the Transformer [4,5]. Following the promising results of the English GEC task, we apply the Transformer with the copying mechanism into the Korean GEC task by introducing novel and effective noising methods for
building Korean GEC datasets. Our contributions are summarized as follows:

- We introduce a novel approach to create Korean GEC datasets by implanting various realistic grammatical errors appearing in Korean texts into original correct sentences and thus capable of creating Korean parallel corpora for GEC in an effective manner.
- We implemented a Transformer-based Korean GEC engine equipped with the copying mechanism and a realistic grammatical error detection and correction rule set for many errors that cannot be handled by the main model.
- We showed that the proposed system drastically outperforms two commercial GEC engines in various aspects.

2 RELATED WORK

Recently, many studies have been conducted on grammatical error correction models based on neural machine translation [1]. The early stages of the research on the NMT-based GEC mainly focused on LSTM-based encoder/decoder [6]. The introduction of the attention mechanisms into the sequence-to-sequence models [7] improves the performance of the GEC [8].

With Transformer [9] actively exploited in many NLP areas, the recent NMT-based GEC approaches are now adapting the Transformer instead of the traditional RNN-based encoder-decoder models and enjoying their competitive and promising performance compared to the conventional architectures [10,11]. The copying mechanism introduced for the machine translation for preserving unknown and special words appeared in source sentences [4] was applied to the GEC models and showed the improved performance in ACL BEA 2019 [12].

Various statistical methods have been studied for constructing parallel corpora for the Korean GEC [13]. The current studies of the NMT-based GEC for Korean language are severely suffering from the lack of the necessary parallel corpora, which makes it very difficult to develop and improve their systems unlike the English GEC. Recently, grammatical noise implantation methods are facilitating the automatic construction of the parallel corpora for the Korean GEC while there is no systematic and effective approach to the noising models specialized for Korean language. Several recent initial attempts are now trying to build the parallel corpora and utilize the Transformer for Korean GEC [3,14].

3 METHOD

3.1 Grammatical Noise Implantation for Korean Language

3.1.1 Grapheme to Phoneme Noising Rules. The complicated pronunciation rules for Korean language lead to the radical and clear difference between its written texts and their pronunciations. The phenomenon causes various lexical errors when writing Korean sentences. One of the pronunciation rules causing errors is “linking sound rule”. The linking sound rule is a phonological phenomenon in which the ending sound of the preceding syllable becomes the first sound of the latter syllable when a syllable that ends with a consonant is followed by a formal morpheme that begins with a vowel [15]. Normally, many people make a mistake by confusing the right words and sentences with their pronunciation, especially produced by the linking sound rule as shown in Table 1.

| Noise (O/X) | Word & Means |
|------------|--------------|
| X          | Pronunciation olaenman-e |
|            | Meaning After a long time |
|            | Korean ‘오랜만에’ |
| O          | Pronunciation olaen-man |
|            | Meaning - |

The noise rules were constructed by using G2PK [16] that can automatically generate Korean spell errors applying the above pronunciation rules. Table 2 shows a Korean sentence generated by the G2PK, in which the correct word, ‘밥을’ is pronounced as ‘bab-eul’ and the incorrect (noised) word, ‘바른’ is sounded as ‘babeul’ artificially generated by the G2PK.

| Type          | Sentence & Means |
|---------------|------------------|
| Original Sentence | naneun eoje bab-eul meog-eosda. |
| Pronunciation | I ate yesterday. |
| Meaning       | ‘밥을 먹었다’. |
| Korean        | ‘나는 어제 떡을 먹었다’. |
| Noised Sentence | naneun eoje bab-eul meog-eosda. |
| Pronunciation | ‘나는 어제 밥을 먹었다’. |
| Meaning       | ‘바른’ |

3.1.2 Heuristic-based Noising Rules. Korean language is morphologically agglutinative, and a word is composed of its component morphemes. Moreover, a single syllable typically consists of an initial, medial and final consonant, which complicates the entire language system even more. These complications cause many people using Korean as their mother tongue to make various mistakes in writing texts. Also, like other languages, Korean is also changing continuously in that newly coined words are created, and its grammatical system is also modified reflecting the current linguistic environment.

In this paper, we aggregate various linguistic errors frequently made by Korean people such as finality errors, ending errors, adverb errors and dialectal confusion. In order to do that, we referred to the Korean grammar correction guidelines and other...
Korean Grammatical Error Correction based on Transformer with Copying Mechanisms and Grammatical Noise Implantation Methods

related documents released by National Institute of Korean Language\(^1\).

| Function Name | Noise (O/X) | Word & Means |
|---------------|-------------|--------------|
| Finality Error | Korean X Pronunciation | ‘오랜만에’ |
|               | Korean X Meaning | After a long time |
|               | Korean O Pronunciation | ‘오랜마에’ |
|               | Korean O Meaning | ‘ والن다’ |
| Verb Confusion Error | Korean X Pronunciation | natda |
|               | Korean X Meaning | Better, recover, get well |
|               | Korean O Pronunciation | natta |
|               | Korean O Meaning | give birth, produce, bear |

3.1.3 Word Spacing Noising Rules. In order to deal with word spacing errors, we also generate word spacing noises by using ChatSpace \(^2\). ChatSpace is an automatic Korean word spacing package while its performance is not so good in practice as you can see in Table 4.

Table 3: Example of Heuristic-based Noising Rules

Table 4: Output of ChatSpace w.r.t. the Input Sentence

| Type            | Sentence & Means             |
|-----------------|------------------------------|
| Original Sentence | 나는 그럴 수 없지.  
| Pronunciation   | naneun geuleolu eobsji.   |
| Meaning         | I cannot do that.  
| Korean          | 나는 그럴 수 없지.  
| Pronunciation   | naneun geuleol su eobsji.  |
| Meaning         | -                           |

We exploit the imperfect behavior of the ChatSpace\(^2\). First of all, an input sentence is passed through the ChatSpace model with all spaces removed. ChatSpace should perform the word spacing with the input and make some mistakes in the process. We consider these mistakes as the word spacing noises.

3.2 Transformer

Our system is based on the attention-based Transformer architecture in which has an encoder and decoder as atomic modules. Each encoder and decoder consist of a multi-head self-attention layer followed by a position-wise feed-forward layer, along with residual connection and layer normalization \(^9\). Unlike the encoder, decoder consists of a total of 3 sub-layers, two of which are the same as the encoder's sub-layer, and the other is a sub-layer that calculates multi-head attention for the output of the encoder. Transformer input embedding is combined with a positional embedding and the token embedding in the input sequence.

3.3 Copying Mechanism

Copying mechanism has proven to be effective for text summarization and semantic parsing \(^18\). Copying mechanism is added to the end of the Transformers \(^4\). The output probability distribution of the copying mechanism is a mixture of \(p^\text{gen}\) and \(p^\text{copy}\). \(p^\text{gen}\) is distribution generated from the decoder. \(p^\text{copy}\) is copy distribution, which is defined as the layer of copy attention that assigns a distribution for tokens that appear in the input sentence. \(a_t^\text{copy}\), which plays the most important role in the copying mechanism, defined per each decoding step. \(a_t^\text{copy}\) is a balance factor that decides whether to reflect the distribution of the input sentence or the distribution generated by the Transformer. And its calculated through the copy scores \(A_t^\text{copy}\), which is output of the copy attention, and the value \(V\) of the copy attentions hidden state.

\[
    a_t^\text{copy} = \text{sigmoid}(W^T \sum(A_t^\text{copy} \cdot V)) 
\]

\[
    P(y_t) = ( 1 - a_t^\text{copy} ) \cdot p^\text{gen}(y_t) + a_t^\text{copy} \cdot p^\text{copy}(y_t) 
\]

As shown in the formula above, if the \(a_t^\text{copy}\) value is greater than 0.5 it reflects copy distribution more in the final distribution value and if it is less than 0.5, it reflects generation distribution. The finally computed distribution determines the word with a high probability as the word in the output sentence \(^4\).

4 EXPERIMENTS AND DISCUSSION

4.1 Data

By applying the previously mentioned noising rules, we constructed a parallel dataset for the Korean GEC by using AI-Hub Korean-English parallel corpus released by NIA \(^3\). The dataset includes 1.1 million Korean-English literary-style sentence pairs and 500K colloquial sentence pairs. Table 5 shows the detailed information of the dataset.

The dataset includes 1,600,000 sentences from various domains such as news articles, web pages, formal documents and even daily conversations, which reflects broad linguistic aspects.

---

\(^1\) https://www.korean.go.kr
\(^2\) https://github.com/pingpong-ai/chatspace
\(^3\) https://www.nia.or.kr
We applied the grammatical noise implantation rules into the dataset and generated a large set of sentence pairs for the Korean GEC.

For the experiments, we generated 16,188,092 sentence pairs of noise implanted sentences and original ones. 12,525,994 pairs were used for the training set and 3,578,738 and 83,460 pairs were used for the development set and test set respectively.

![Model Architecture](image)

**Figure 1: Model Architecture**

**Table 5: Statistics and Elements of the Dataset Used**

| Domain          | Explanation                                      | Size  |
|-----------------|--------------------------------------------------|-------|
| News            | News text                                        | 800K  |
| Government      | Government and Local Government Websites, Publications | 100K  |
| Law             | Administrative rules, autonomous laws            | 100K  |
| Korean Culture  | Korean history and culture contents              | 100K  |
| Colloquial      | Natural colloquial sentences                     | 400K  |
| Dialogue        | Context/scenario-based conversation set          | 100K  |

**4.2 Model and Parameters**

Our GEC model uses a typical configuration of the Transformer with the copying mechanism in that all the input tokens are embedded and encoded by the conventional positional encoding mechanism. We use a 4,096-dimensional position-wise feed-forward layer. In addition, both the token embedding size and hidden size are 512. As for the copying mechanism, we apply a single layer with 8 attention heads. Adam optimizer was used in the training. The batch size during training was set to 100 and the dropout ratio and label smoothing value were all set to 0.1. We trained our own tokenizer by using SentencePiece [19] where the size of the source (encoder) and target (decoder) vocabulary is set to 30,000.

**Table 6: Parameters Size**

| Parameters                        | Size |
|-----------------------------------|------|
| Position-wise Feed forward layer  | 4096 |
| Encoder/Decoder Layer size       | 8    |
| Embedding Size                    | 512  |
| Attention-Head                    | 8    |
| Dropout ratio                     | 0.1  |
| Smoothing value                   | 0.1  |
| Vocabulary size                   | 30,000 |
4.3 Evaluation Metrics

We compared the performance of our system with the py-hanspell (Naver API)[20] and hanspell (Kakao API)[21] published on the web. The performance of the system was evaluated by both GLEU [22] and BLEU [23] scores. BLEU score is frequently used for evaluating machine translation models by measuring the similarity of correct translations and system outputs. The GLEU metric is a variant of BLEU proposed for evaluating grammatical error corrections using n-gram overlap with a set of reference sentences, as opposed to precision/recall of specific annotated errors [24].

4.4 Results and Discussion

Table 7 shows the comparative results of the proposed system and the two commercial grammar checking systems by using both BLEU and GLEU scores.

Table 7: Comparison of GEC Models (GLEU, BLEU)

| Model         | GLEU  | BLEU  |
|---------------|-------|-------|
| Py-Hanspell[20] | 76.68 | 75.68 |
| Hanspell[21]   | 77.08 | 76.68 |
| Ours           | 88.71 | 88.01 |

As seen in the Table 7, our system outperforms all the grammatical checking services in all metrics. In particular, the GLEU score of our system is 88.71, which shows the superiority of the system in the grammatical error correction performance compared to the other two systems (+12.03 and +11.63).

Table 8: Comparison of GEC Models (Precision, Recall, F0.5)

| Model         | Precision | Recall | F0.5  |
|---------------|-----------|--------|-------|
| Py-Hanspell[20] | 36.01     | 26.05  | 29.62 |
| Hanspell[21]   | 37.44     | 27.36  | 32.35 |
| Ours           | 62.03     | 61.52  | 61.72 |

Table 8 shows the detailed evaluation results of the systems denoting precision, recall and F0.5 scores by using our test data mentioned earlier. Our grammatical noise implantation method mainly reflects typically and frequently committed grammatical errors that all the conventional grammar checking, and correcting systems should handle effectively. Therefore, the comparison using the test set seems to be fair and objective. The superior performance of the model proposed in this paper was more clearly demonstrated in precision, recall, and F0.5 scores. The model in this paper has a precision of 62.03, which is about 25 higher than other models. In Recall, 61.52 and F0.5 are 61.72, which is more than 30 different from other models, showing high performance.

Table 9: Error Correction Results by the Three Systems

| Input Sentence                      | Py-Hanspell (Kakao API) | Hanspell (Kakao API) | Ours (Naver API) |
|-------------------------------------|-------------------------|----------------------|------------------|
| 수업 (taught) 이수학을 가르쳤다.  (The teacher taught math in class.) | 수업 시 가네 선생님이 수학을 가르쳤다. | 수업 시간에 선생님이 수학을 가르쳤다. | 수업 시간에 선생님이 수학을 가르쳤다. |

Table 9 denotes the outputs of the three systems used in the experiment with an input sentence with various grammatical errors including a pronunciation-related error, contextual error and word spacing error. In the sentence, the pronunciation-related error is denoted in italic, the contextual error is indicated by bold-face and the word spacing error is marked by underscore. While py-hanspell (Naver API) could detect and correct the second word spacing error, it fails to handle all the others. In particular, py-hanspell (Naver API) incorrectly revised the first word spacing error suggesting an overly spaced tokens. Besides, hanspell (Kakao API) fails to handle all the errors in the sentence. On the contrary, our system successfully detect, and correct all the errors in the sentence. In particular, our system could detect and correct the contextual error by revising the word '가르쳤다' (pointed to) which is lexically correct but inappropriate semantically to '가르쳤다 (taught)'.

5 CONCLUSIONS

This paper introduced a Korean GEC model based on Transformers equipped with a copying mechanism as well as a systematic process for automatically constructing parallel corpus for the proposed model. The process involves three grammatical noise implantation rules reflecting general linguistic mistakes made in writing Korean texts. The experimental results indicated that the proposed system outperforms the existing commercial grammar check and correction services in many perspectives including GLEU, BLEU, Precision, Recall and F0.5.

Although we attempted to apply typical and frequent errors and typos in generating our dataset, we still seem to be light on the noising rules covering other grammatical mistakes and
semantic misuses in Korean language. Therefore, our future research direction would be the enlargement of the rule set by more intensively inspecting error patterns. By applying the extended rule set, it is necessary to construct more expressive datasets covering almost all the lexical, syntactic and semantic errors appeared in Korean texts.

ACKNOWLEDGMENTS
This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (2018R1D1A1B07048839)

REFERENCES
[1] Wang, Y., Wang, Y., Liu, J., & Liu, Z. 2020. A Comprehensive Survey of Grammar Error Correction. arXiv preprint arXiv:2005.06600.
[2] Bak, S. H., Lee, E. J., & Kim, P. K. 2017. A Method for Spelling Error Correction in Korean Using a Hangul Edit Distance Algorithm. Smart Media Journal, 6(1), 16-21.
[3] Cho, S. W., Kwon, H. S., Jung, H. Y., Lee, J. H. 2018. Adoption of a Neural Language Model in an Encoder for Encoder-Decoder based Korean Grammatical Error Correction. KIIS Transactions on Computing Practices, 24(6), 301-306
[4] Zhao, W., Wang, L., Shen, K., Jia, B., & Liu, J. 2019. Improving grammatical error correction via pre-training a copy-augmented architecture with unlabeled data. arXiv preprint arXiv:1903.00138.
[5] Bryant, C., Felice, M., Andersen, Ø. E., & Briscoe, T. 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, August 2019, 23 pages.
[6] Yuan, Z., & Briscoe, T. 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, 380-386.
[7] Bahdanau, D., Cho, K., & Bengio, Y. 2014. Neural machine translation by jointly learning to align and translate. arXiv preprint arXiv:1409.0473.
[8] Xie, Z., Avati, A., Arivazhagan, N., Jurafsky, D., & Ng, A. Y. 2016. Neural language correction with character-based attention. arXiv preprint arXiv:1603.09727.
[9] Vaovani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., & Polosukhin, I. 2017. Attention is all you need. In Advances in neural information processing systems. 5998-6008.
[10] Lichtarge, J., Alberti, C., Kumar, S., Shazeer, N., & Parmar, N. 2018. Weakly supervised grammatical error correction using iterative decoding. arXiv preprint arXiv:1811.01710.
[11] Chollampatt, S., & Ng, H. T. 2018. A multilayer convolutional encoder-decoder neural network for grammatical error correction. arXiv preprint arXiv:1801.08831.
[12] Choe, Y. J., Ham, J., Park, K., & Yoon, Y. 2019. A neural grammatical error correction system built on better pre-training and sequential transfer learning. arXiv preprint arXiv:1907.01256.
[13] Lee, J. H., Kim, M. h., Kwon, H. C. 2017. The Utilization of Local Document Information to Improve Statistical Context-Sensitive Spelling Error Correction. KIIE Transactions on Computing Practices. 23. 446-451
[14] Park, C. J., Jeong, S. Yang, K. S., Joe, J. C & Lim, H. S. 2019. Korean Spell Correction based on Denoising Transformer. The 31st Annual Conference on Human & Cognitive Language Technology (daejeon, Korea). Korean Institute of Information Scientists and Engineers, 368-374.
[15] Kim, S. K., Kime, T. Y., Kang, R. W., Kim, J. 2020. Characteristics of Korean Liaison Rule in the Reading and Writing of Children of Korean-Vietnamese Multicultural Families and the Correlation With Mothers’ Korean Abilities. Korean Speech-Language & Hearing Association (KSHA) 29, 3 (April 2020), 57-71. DOI: https://doi.org/10.15772/jslh.2020.29.2.057
[16] Park, K. B., 2019. g2pk. Github repository: https://github.com/Kyubong/g2pk
[17] Seo, S. I., 2020. ChatSpace: Space Correcting Model for Improving Tokenization of Korean Conversational Text. Master’s thesis. Yonsei University, Seoul, Republic of Korea.
[18] Robin Jia and Percy Liang. 2016. Data recombination for neural semantic parsing. arXiv preprint arXiv:1606.03622.
[19] Kudo, T., & Richardson, J. 2018. Sentencepiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. arXiv preprint arXiv:1808.06226.
[20] Py-Hanspell(2020), https://github.com/ssut/py-hanspell(2020.07.15)
[21] Hanspell(2020), https://github.com/9beach/hanspell(2020.07.15)