Grammatical Error Correction (GEC) of Indonesian Text Based on Neural Machine Translation (NMT)

1st Nike Sartika
Department of Electrical Engineering
UIN Sunan Gunung Djati Bandung line Bandung, Indonesia
nikesartika@uinsgd.ac.id

2nd Yuda Sukmana
Sekolah Teknik Elektro dan Informatika Institut Teknologi Bandung
Bandung, Indonesia
yudasukmana@itb.ac.id

*Corresponding author: nikesartika@uinsgd.ac.id
Received: September 15, 2023; **Accepted**: November 27, 2023

***Abstract*—Writing errors in Indonesian are often found in various writings made in educational, government and mass media environments. The most dominant error is in spelling. This research proposes a Grammatical Error Correction (GEC) for Indonesian using the Neural Machine Translation (NMT) method, namely seq2seq, which is popularly used for English and has achieved the best performance approaching human capabilities. The model developed is made into a web-based service that is easy for users to access. The datasets used in this experiment are artificial datasets sourced from several studies regarding error analysis in Indonesian. The research results show that with the help of currently available open-source tools such as OpenNMT-py, it is possible to simplify the training process of NMT-based GEC models. Unfortunately, the small number of datasets leads to poor predictions for random sentences.**

Keywords—grammatical error correction, neural machine translation, Indonesian text, seq2seq.

I. INTRODUCTION

Writing errors in Indonesian are often found in various writings made in educational, government and mass media environments. Research conducted by Purwandari et al. [1] in the Jladri Village government environment found that the most dominant error in writing official letters was spelling. Another research conducted in the Kendari City government environment by Sukmawati et al. [2] also showed similar errors, namely spelling errors in general and commercial information services. Several studies conducted in the world of education also show the same thing, spelling errors are language errors that are often found in written works made by students at school such as essays and negotiation texts [3-5] and scientific papers created by students at universities such as papers [6-9] Apart from that, spelling errors were also found in articles published in the mass media based on research conducted by [10].

These findings are an indicator that writing skills, especially the use of spelling, are language skills that need to be given more attention because there are still many errors found. Unfortunately, there are currently no tools that can detect or correct these errors automatically. Previously, research conducted by [11] proposed a proofreading tool for Indonesian texts using statistical and rule-based models. Statistical Machine Translation (SMT) is a technique that is widely used for machine translators like this, including translating from incorrect text to correct text or often called Grammatical Error Correction (GEC). However, according to [12] SMT has weak capabilities in terms of generalization and capturing global dependencies because it requires defining rules manually and large amounts of dictionary data [13].

Research on the topic of GEC currently proposes the use of NMT or Neural Machine Translation methods to overcome this and has been proven to have good effects. Unfortunately, many of these studies focus on English, so there has not been much research on the use of NMT for Indonesian GEC. In this research, a GEC model will be proposed for Indonesian using the NMT method, namely seq2seq, which is popularly used for English and has achieved the best performance (state-of-the-art performance) approaching human capabilities [14]. The model developed will be made into a web-based service that is easy for users to access.

NMT research for Indonesian spelling was carried out by [13] they used Long Short-Term Memory (LSTM). The proposed solution is in two stages, the first is detecting spelling errors and the second is correcting the errors, this stage is similar to what was done by [12]. This stage begins with a preprocessing step by carrying out tokenization and POS (Part-of-speech) tagging. Spelling errors are detected by searching (lookup) in the Big Indonesian Dictionary (KBBI), if a word is found it will be passed to the LSTM model as a model for correcting spelling, if the output from LSTM is not found in the dictionary then the initial word will be the final output and vice versa . If from the initial detection, a word is not found in the dictionary, it will be directly passed to the LSTM model and whatever output it produces will be the final output. As a result, they got an accuracy of up to 83.76%.

II. METHODS

This research will focus on GEC for Indonesian language texts using the LSTM-based seq2seq model. The seq2seq model consists of an encoder and a decoder, the input goes to the encoder to be processed then produces a vector (context) which will go to the decoder to produce output. One drawback of the seq2seq model is that the initial part of the input sequence is often forgotten after processing completes the entire input [15]. However, this can be overcome by

Vol. 05 No. 2, October-2023, Pages 66-69, DOI : https://dx.doi.org/10.20961/jeeict.5.2.78837

adding an attention mechanism to the seq2seq model. An illustration of this model can be seen in Figure 1.

GEC can basically be treated like a translator, namely a translator from bad sentences to good sentences. It is impossible to correct all grammatical errors in a sentence with one round [12] so the Recycling Generating method introduced by [14] will be used, where sentences are corrected using a multi-round technique.. First, the original sentence X is entered into the seq2seq model producing a corrected output in the form of sentence This round continues until the probability of sentence Xt is greater than Xt-1. This round process can be seen in Figure 2 [14].

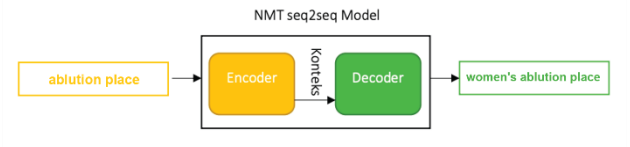

Fig. 1. NMT architecture seq2seq model.

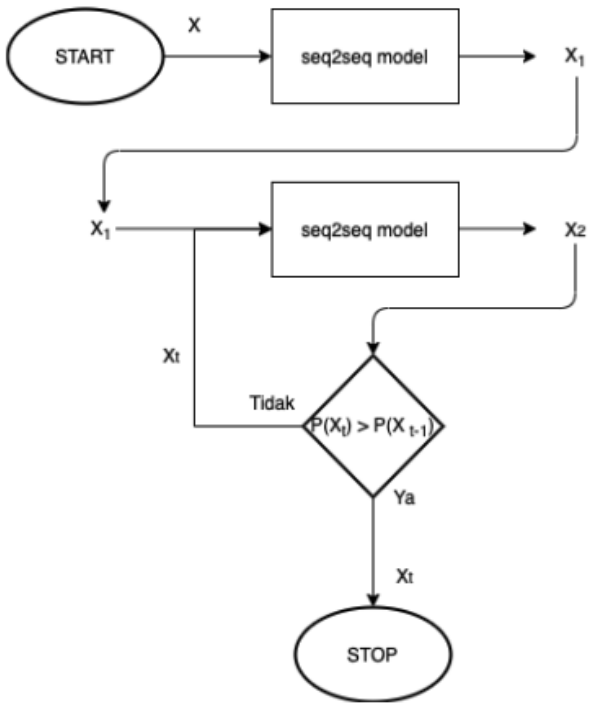

Fig. 2. Recycling generating.

Because there are not many datasets to be used, there is a possibility that the seq2seq model will correct the sentence incorrectly so that the output sentence is of worse quality than the original sentence. To overcome this, [12] proposed a re-ranking technique by making the sentence with the highest probability the output. The re-ranking formula with the highest probability is shown in equation (1) [12].

$$y = (p(X_t), p(X)) \quad (1)$$

III. Result and Discussion

A. Datasets

The datasets used in this experiment are artificial datasets sourced from several studies regarding error analysis in Indonesian which were obtained through searching for scientific articles on Google Scholar using the keyword "analysis of errors in Indonesian writing" and filtering articles published since 2017, with search results as many as 15,300 articles. In this type of research, errors are usually analyzed in written works such as scientific papers, newspapers, textbooks, and so on, then the researcher corrects the errors.

The results of the analysis and correction will become the basis for creating artificial datasets in the form of error corrected datasets. Specifically for the experiment carried out this time, researchers will limit users of the data search results to articles that appear on the first 4 pages of Google Scholar, namely 40 articles. From these 40 articles, 430 Indonesian language error-corrected datasets were obtained which will be divided into 400 training datasets and 30 evaluation datasets.

Apart from that, to produce a model with better output, pretrained word embeddings from fastText will be used. Word embeddings are vector representations of Indonesian words in alphanumeric form [13].

B. Model and Training

In this experiment, the Transformer seq2seq model will be implemented using OpenNMT, which is an open source neural machine translation first developed by Yoon Kim from Harvard NLP. The OpenNMT used requires a minimum of PyTorch version 1.6.0 and Python version 3.6. This training model will be carried out in an iMac 2012 environment with CPU specifications 2.7 GHz Quad-Core Intel Core i5, 8 GB memory, GPU Nvidia GeForce GT 640M 512 MB. OpenNMT-py can be installed using the pip package manager.

C. Experimental Result

The artificial dataset that has previously been created will be created into several files as shown in Table 1.

The dataset will be created into a vocabulary list. To be able to do this, we can run the script from OpenNMT-py, but first we have to create a configuration file for this project which will be called gce_id.yaml. The configuration file of OpenNMT-py gce_id.yaml is shown in Figure 3.

Next, you can run the script onmt_build_vocab -config gce_id.yaml -n_sample 400 in the Terminal in the project directory where the gce_id.yaml and dataset files are located. The result is 1183 vocabularies in the src file and 1163 in the tgt file, these vocabularies are stored in the gce.vocab.src and gce.vocab.tg files in the data/run directory.

After the vocabulary has been successfully created, the next step is to carry out model training. This training model will use CPU, namely a model consisting of 2-layer LSTM with 500 hidden units.

To be able to carry out model training, we must add configuration to the gce_id.yaml file to determine the location of the model storage file, number of trains_steps, and valid_steps. After that, you can run the OpenNMT-py script to carry out training, namely: onmt_train -config gce_id.yaml in the project directory. This training process will take a long time, especially if you only use the CPU. This training was completed in 942 seconds, acc: 42.83, ppl: 14.27, and 476/515 tok/s. To test this model we will create a src-test.txt file which contains incorrect Indonesian sentences, then use OpenNMT-py to translate, namely: onmt_translate -model data/run/model_step_1000.pt -src data/src-test.txt -verbose , the result is that the average prediction score is -0.2012,

which is the closer to 0, the better the prediction and the prediction ppl (perplexity) is 1.2229. From the prediction results, it can be said that this model is quite good at predicting sentences in the dataset, but if what is tested is a random sentence then the results are very bad. Figure 4 shows an example of GCE test results with the model that has been created.

TABLE I. DATASET FILES

File Name	Description
src-train.txt	Contains 400 Indonesian sentences with writing errors
src-trgt.txt	Contains 400 Indonesian sentences whose errors have been corrected
src-val.txt	Contains 30 Indonesian sentences with writing errors
src-tgt.txt	Contains 30 Indonesian sentences whose errors have been corrected

```
gce_id.yaml
## Where the samples will be written
save_data: data/run/gce
## Where the vocab(s) will be written
src_vocab: data/run/gce.vocab.src
tgt_vocab: data/run/gce.vocab.tgt
# Prevent overwriting existing files in the folder
overwrite: False

# Corpus opts:
data:
    corpus_1:
        path_src: data/src-train.txt
        path_tgt: data/tgt-train.txt
    valid:
        path_src: data/src-val.txt
        path_tgt: data/tgt-val.txt

```

Fig. 3. Configuration file of OpenNMT-py gce_id.yaml

```
[2021-04-20 23:53:12,568 INFO]
SENT 16: ['di', 'tuliskan']
PRED 16: dituliskan
PRED SCORE: -0.0311

[2021-04-20 23:53:12,568 INFO]
SENT 17: ['di', 'rekomendasikan']
PRED 17: direkomendasikan
PRED SCORE: -0.0691

[2021-04-20 23:53:12,568 INFO]
SENT 18: ['di', 'tulis']
PRED 18: ditulis
PRED SCORE: -0.0545

[2021-04-20 23:53:12,568 INFO]
SENT 19: ['di', 'lakukannya']
PRED 19: dilakukannya
PRED SCORE: -0.0856

[2021-04-20 23:53:12,568 INFO]
SENT 20: ['didalam']
PRED 20: di dalam
PRED SCORE: -0.0019

[2021-04-20 23:53:12,568 INFO]
SENT 21: ['Keterangan', 'kode', 'nomer']
PRED 21: Keterangan kode nomor
PRED SCORE: -0.2483

[2021-04-20 23:53:12,568 INFO]
SENT 22: ['Bahkan', 'ada', 'sebagian', 'lahan', 'persawahan', 'yang', 'sudah', 'menebar', 'benih']
PRED 22: Bahkan ada sebagian lahan persawahan yang sudah ditebar benih
PRED SCORE: -0.9881

[2021-04-20 23:53:12,569 INFO] PRED AVG SCORE: -0.0652, PRED PPL: 1.0673
```

Fig. 4. GCE test results.

D. Model Implementation

The model that has been trained will be deployed and can then be accessed via the HTTP API. A web-based user interface will be created to make it easier for users to correct Indonesian text using the model that has been created. The mockup design of the web page can be seen in Figure 5.

This implementation will use the server.py module from OpenNMT-py and then CORS support (flask_cors) will be added so that it can be accessed directly from browsers with different domains. This module must be run by calling the command: python server.py, then to test the HTTP API of the OpenNMT-py translation server module, CURL will be used with the POST method to the URL http://0.0.0.0:5000/translator/translate as shown in Figure 6.

The web application was developed using the React JavaScript UI Library with Material-UI, and Axios to access the HTTP API to perform previously created translations. Figure 7 shows the results of implementing a web application for correcting Indonesian texts using the NMT model that has been developed.

Fig. 5. Web-based model usage implementation page mockup.

```
curl -i -X POST -H "Content-Type: application/json" \
    -d '[{"src": "didalam", "id": 100}]' \
    http://0.0.0.0:5000/translator/translate
HTTP/1.1 200 OK
Access-Control-Allow-Origin: *
Content-Length: 86
Content-Type: application/json
Date: Tue, 20 Apr 2021 17:01:03 GMT
Server: waitress

[[{"n_best":1,"pred_score":-0.0019094559829682112,"src":"didalam","tgt":"di dalam"}]]
```

Fig. 6. HTTP API test results using the curl command in Terminal.

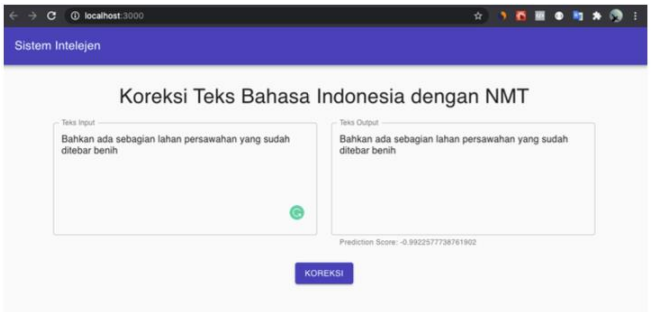

Fig. 7. Web application implementation.

IV. CONCLUSION

The experiment of creating a Grammatical Error Correction project for Indonesian texts has been successfully carried out. With the help of currently available open source tools such as OpenNMT-py, the training process of NMT-based GEC models is simplified. However, there are still many shortcomings, such as the small number of datasets, which causes poor predictions for random sentences. The next work is to train the model using a larger dataset so that a GEC can be created that is able to correct errors well and thoroughly.

REFERENCES

[1] Purwandari, H. S., Setiawan, B., dan Saddhono, K., "Analisis Kesalahan Berbahasa Indonesia pada Surat Dinas Kantor Kepala Desa Jladri", BASASTRA Jurnal Penelitian Bahasa, Sastra Indonesia dan Pengajarannya, Vol. 1, No. 3, 478–489, April 2014.

[2] Sukmawati, Nurhayati, dan Iswary, E, "Penggunaan Bahasa Indonesia pada Informasi Layanan Umum dan Layanan Niaga di Kota Kendari", Jurnal Bahasa dan Sastra, 2(1), 3–4, 2013.

[3] Ariningsih, N., Sumarwati, S., dan Saddhono, K., "Analisis Kesalahan Berbahasa Indonesia Dalam Karangan Eksposisi Siswa Sekolah Menengah Atas", Jurnal Penelitian Bahasa, Sastra Indonesia, dan Pengajarannya, 1(1), 130–141, 2012.

[4] Khoirurrohman Taufiq, "Analisis Kesalahan Ejaan Dalam Karangan Siswa Kelas 3 Sdn Ketug Kecamatan Butuh Tahun Pelajaran 2017/2018", Jurnal Dialektika Jurusan PGSD, 8(2), 70–77, 2018.

[5] Qhadafi, M. R., "Analisis Kesalahan Penulisan Ejaan yang Disempurnakan dalam Teks Negosiasi Siswa SMA Negeri 3 Palu", Jurnal Bahasa dan Sastra, 3(4), 1–21, 2018.

[6] Asih, A., Tantri, S., dan Sutresna, I. B., "Kesalahan Penggunaan Ejaan Bahasa Indonesia dalam Makalah sebagai Alternatif Materi Ajar Ejaan Bahasa Indonesia (EBI)", Prosiding Seminar Nasional V: Bahasa, Sastra, dan Pengajarannya, diperoleh melalui situs internet: https://eproceeding.undiksha.ac.id/index.php/semnasbasindo, 191–199. 2018.

[7] Leksono, M. L., "Analisis Kesalahan Penggunaan Pedoman Ejaan Bahasa Indonesia (PUEBI) Pada Tugas Makalah dan Laporan Praktikum Mahasiswa IT Telkom Purwokerto", JP-BSI (Jurnal Pendidikan Bahasa dan Sastra Indonesia), 4(2), 116. https://doi.org/10.26737/jp-bsi.v4i2.1106. 2019.

[8] Rosdiana, L. A., "Kesalahan Penggunaan Ejaan Bahasa Indonesia (EBI) Pada Karya Ilmiah Mahasiswa, Bahtera Indonesia; Jurnal Penelitian Bahasa dan Sastra Indonesia, 5(1), 1–11. https://doi.org/10.31943/bi.v5i1.58, 2020.

[9] Turistiani, T. D., "Fitur Kesalahan Penggunaan Ejaan Yang Disempurnakan Dalam Makalah Mahasiswa, Paramasastra", 1(1), 61–72. https://doi.org/10.26740/parama.v1i1.1470, 2014.

[10] Winata, N. T., "Analisis Kesalahan Ejaan Bahasa Indonesia Dalam Media Massa Daring (Detikcom), Bahtera Indonesia", Jurnal Penelitian Bahasa dan Sastra Indonesia, 4(2), 115–121. https://doi.org/10.31943/bi.v4i2.52, 2019.

[11] Fahda, A., dan Purwarianti, A., "A statistical and rule-based spelling and grammar checker for Indonesian text", Proceedings of 2017 International Conference on Data and Software Engineering, ICoDSE 2017, 2018-January, 1–6. https://doi.org/10.1109/ICODSE.2017.8285846, 2018.

[12] Qiu, Z., dan Qu, Y., "A Two-Stage Model for Chinese Grammatical Error Correction", IEEE Access, 7, 146772–146777. https://doi.org/10.1109/ACCESS.2019.2940607, 2019.

[13] Zaky, D., and Romadhony, A., "An LSTM-based Spell Checker for Indonesian Text", Proceedings - 2019 International Conference on Advanced Informatics: Concepts, Theory, and Applications, ICAICTA 2019, 1–6. https://doi.org/10.1109/ICAICTA.2019.8904218, 2019.

[14] Ge, Tao, Furu Wei, and Ming Zhou., "Reaching human-level performance in automatic grammatical error correction: An empirical study." arXiv preprint arXiv:1807.01270 (2018).

[15] Y. Heryadi, B. D. Wijanarko, D. F. Murad, C. Tho and K. Hashimoto, "Neural Machine Translation Approach for Low-resource Languages using Long Short-term Memory Model," 2023 International Conference on Computer Science, Information Technology and Engineering (ICCoSITE), Jakarta, Indonesia, 2023, pp. 939-944, doi: 10.1109/ICCoSITE57641.2023.10127724.