Electronic Journal of Education, Social Economics and Technology
Vol. 5, No. 1, (2024), pp. 30~39
ISSN 2723-6250 (online)
DOI: https://doi.org/10.33122/ejeset.v5i1.309

SAINTIS publishing

Research Article

LSTM-Based NLP Approach for Spelling Error Detection and Correction in Scientific Writing Indonesian Language

Yeru Dwi Pratama Halim[1*], and Ida Nurhaida[1,2*]

[1] Department of Informatics, Universitas Pembangunan Jaya, South Tangerang, Indonesia, 15413
[2] Center for Urban Studies, Universitas Pembangunan Jaya, South Tangerang, Indonesia, 15413
*Corresponding Author: ida.nurhaida@upj.ac.id | Phone Number: +6287876725326

ABSTRACT

Scientific writing requires precision and clarity to uphold credibility and effective communication. Errors such as spelling mistakes and typos can compromise the quality and reliability of scientific texts. This study proposes a Long Short-Term Memory (LSTM)-based approach to detect and correct spelling errors, enhancing text accuracy and readability. The dataset comprises 45,698 standard words, supplemented with typo variations to improve model performance. Data is sourced from the Indonesian Dictionary (KBBI) and undergoes normalization and preprocessing to capture diverse error patterns. The model's performance is evaluated using a confusion matrix, achieving 93% accuracy and high precision, recall, and F1-score metrics. These results demonstrate that the proposed NLP-based LSTM model offers an effective and reliable solution for identifying and correcting spelling errors. This approach significantly enhances the quality of scientific writing, ensuring more transparent and credible communication.

Keywords: Scientific Writing; Natural Language Processing; Text Correction System; LSTM

1. INTRODUCTION

The rapidly advancing digital era has made the exchange of information through written texts increasingly common, especially in the context of scientific writing. Scientific writing is a work that addresses an existing issue and is substantiated by facts and research data (Khaidir et al., 2023). Scientific writing serves a vital function in communicating knowledge and the latest discoveries across various fields of study. Writing scientific papers must adhere to scientific principles such as objectivity, logic, systematic structure, empirical evidence, clarity, and consistency (Rosmiati, 2017). Words play a crucial role in conveying ideas in a scientific work. Therefore, to ensure that readers understand the concepts and messages in a written piece well, automatic text correction plays a vital role in creating a scientific work (Dewi & Qoiriah, 2021).

The mistakes that occur during the creation of scientific work can have serious consequences. As seen in several cases related to writing errors, such mistakes have occurred. In the main news article published in the October 2023 edition, quoted by Suprihatma (2024), there are several examples of articles that experienced errors, such as the misuse of words, writing mistakes, and word redundancy, such as the use of the word “nampak”, The correct word should be “tampak”. Spelling errors also occur on signboards in stores, such as 'Apotik Abila Farma,' where the word 'apotik' is not standard. The correct spelling should be 'Apotek Abila Farma (Marlina, 2019). The next case involves a student who made spelling mistakes in English, as mentioned by Adawiyah (2023), mentions that there are spelling mistakes in the student's writing, such as "very day" which should be "every day," "bathe" which should be "bath," and "carred out" which should be "carried out." Such errors not only cause confusion but can also damage the credibility of the writer or the associated institution. Therefore, efforts to minimize mistakes in writing, including utilizing technologies like automatic correction tools, are essential to enhance the quality and precision of scholarly writing.

Text correction is a crucial aspect in various fields and contexts. In the academic world, text correction ensures that scientific papers, research papers, and dissertations are of high quality and free from errors (Juniarti, 2019). This is important to maintain the credibility and validity of the scientific work. Furthermore, in publishing, text correction is crucial for producing high-quality books, articles, and other written materials free from spelling and grammatical errors (Susanti et al., 2022). This helps maintain the publisher's reputation and ensures reader satisfaction. Furthermore, correction is crucial in the communications and marketing industry to ensure that promotional content, advertisements, and marketing materials are free from errors that could tarnish the brand's image or convey a misleading message (Siregar et al., 2023). In a legal context, text correction ensures that legal documents, contracts, and regulations are written clearly and accurately, avoiding ambiguities or misunderstandings that could have serious consequences. Even in everyday communication, such as emails and instant messages, text correction can help enhance professionalism and clarify the message's intent. Text correction holds crucial function in maintaining the quality and accuracy of written information and ensuring effective communication across various aspects of life (Yulianizar & Waliah, 2022). Based on this, text correction is not merely a technical step but also a crucial process to support professionalism and credibility in various fields.

One of the main challenges that often occurs is the presence of spelling or typographical errors. Spelling or typographical errors are common mistakes writers make when composing scientific papers. These errors can diminish the clarity of the message and affect the reader's understanding (Herawati et al., 2022). A spelling error is a mistake made while typing caused by accidental actions, typing machine errors, or a slip of the finger, as well as the proximity of letters on the keyboard (Suhendar et al., 2023). We often encounter features for automatically correcting spelling errors in word processing software. However, these features are not fully utilized, leading to frequent writing mistakes. Writers usually fail to notice these errors because they focus more on the content of their scientific work than on checking the spelling (Rustanti, 2024). Therefore, there is a need for a Natural Language Processing (NLP) feature to check spelling/writing errors using the LSTM method.

Natural Language Processing (NLP) is a discipline that allows systems to interpret, analyze, and produce text that is understandable to humans (Amien, 2023). In line with this, Rayhan et al. (2023) also mentioned in his research that Natural Language Processing (NLP) is an interdisciplinary area that utilizes machines to comprehend, analyze, and produce human language in a way that is both meaningful and contextually relevant. In Natural Language Processing (NLP), there is a feature that can assist in checking spelling or word writing errors, known as String Matching. Utama et al. (2021), mentions that String Matching refers to searching for a string composed of several words within a text. Algorithms used for string searching include the Hamming Distance, Jaccard Distance, Levenshtein Distance, Damerau-Levenshtein Distance, Jaro Distance, and Jaro-Winkler Distance algorithms. Based on this explanation, Natural Language Processing (NLP) can assist in text correction in academic writing.

Kusuma and Ratnasari (2023), have conducted research on the development of Natural Language Processing (NLP) for text correction. They compared three spelling correction algorithms for the Indonesian language: Peter Norvig, N-Gram, and LSTM. The results showed that the Peter Norvig algorithm had the highest accuracy rate at 89%, followed by the N-Gram method with an accuracy of 75%, and LSTM at 74%. These findings indicate that the Peter Norvig approach is superior in the context of Indonesian spelling checking.

Meanwhile, a study by Yang et al. (2020) explores a text error correction method based on prefix tree merging for the Chinese language. The LSTM algorithm used in this study achieved an accuracy of 73.65% in detecting real-world errors and 72.68% in detecting non-multi-character errors.

Additionally, the study by Zaky and Romadhony (2019), explores using the Long Short-Term Memory (LSTM) model for text error correction in Indonesian. This model achieved an accuracy of 83.76% in detecting incorrect words using a combination of character features, Word2Vec, and Part of Speech (POS) tags from surrounding words. The study focuses on correcting four types of character errors: addition, deletion, substitution, and transformation, with the best results obtained for character deletion errors.

In this study, the researcher explores the development of Natural Language Processing (NLP) using the Long Short-Term Memory (LSTM) algorithm as a text correction tool designed to detect and correct writing errors with high accuracy. Unlike previous studies, the approach taken in this research involves increasing the variation of typos through data augmentation techniques, such as deleting, inserting, and swapping letters within words. This approach aims to enable the model to learn a broader range of typing errors and improve its ability to detect and correct spelling mistakes more accurately. Additionally, to evaluate the model's performance, this study uses an evaluation based on a confusion matrix, allowing the researcher to measure the model's accuracy in greater depth through metrics such as precision, recall, and F1-score.

The research problem formulated in this study is how the NLP approach can assist the system in performing automatic text correction with high accuracy. Therefore, to address this problem, this study aims to develop a system for text correction using the NLP approach to improve word writing in academic papers and to determine the accuracy of the text correction using the developed NLP approach.

2. RESEARCH METHOD

The research method consists of systematic steps to achieve the research objectives, as shown in Figure 1.

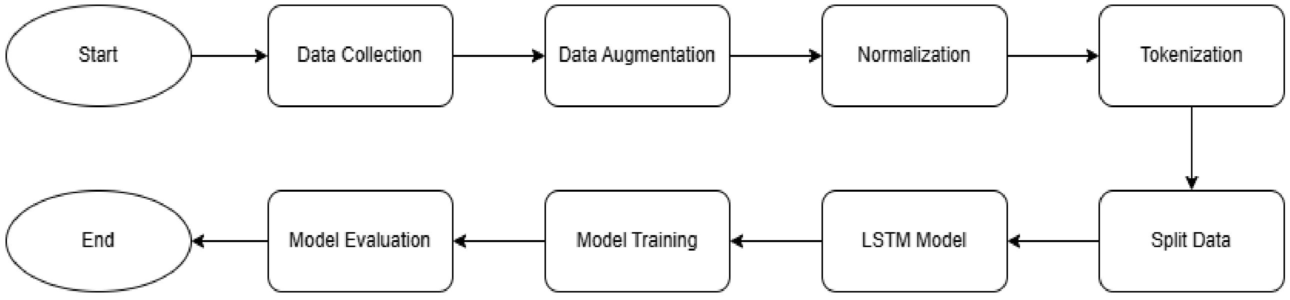

Figure 1. Research Method

2.1. Data Collection

The initial step in this research is to collect standard words from the KBBI platform. The collected data is then processed to ensure completeness, consistency, and relevance to the research needs. Subsequently, this data will serve as the primary reference in developing an automatic text correction model capable of detecting and correcting spelling errors.

2.2. Data Augmentation

Figure 2. Data Augmentation

After collecting the standard word data, the next step is to increase the variation of typos in the dataset. These typo variations are generated by mimicking typical typing mistakes, including substituting letters, omitting characters, adding extraneous symbols, and errors related to letter substitution, insertion, or transposition within words. The generated standard words are used to introduce typo variations for each standard word, enrich the dataset, and represent different potential typing mistakes. By increasing these typo variations, the developed model can learn error patterns more effectively, thereby improving its accuracy in identifying more complex mistakes and enhancing the system's ability to correct text accurately. This process can be seen in Figure 2.

2.3. Normalization

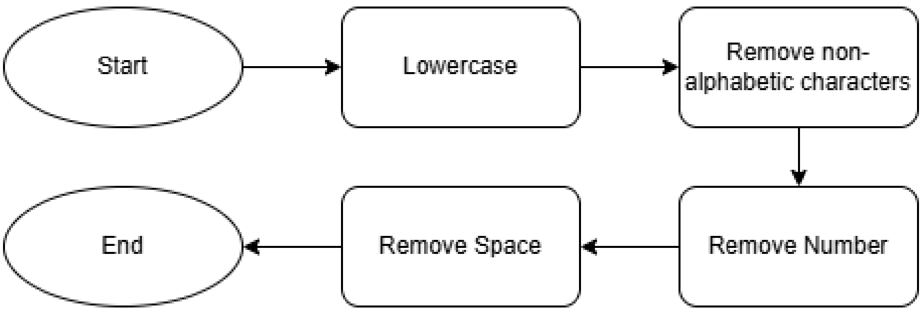

Figure 3. Normalization

After adding variations of typo words into the dataset, the next step is to perform the normalization process to ensure the data is consistent and ready for model training. The normalization process includes several steps, such as converting all letters to lowercase, removing irrelevant symbols or characters, eliminating numbers, and ensuring no duplicate or redundant data (Putri & Cahyono, 2024). This step aims to enhance data quality and minimize potential errors during subsequent processing stages. This process is illustrated in **Figure 3**.

2.4. Tokenization

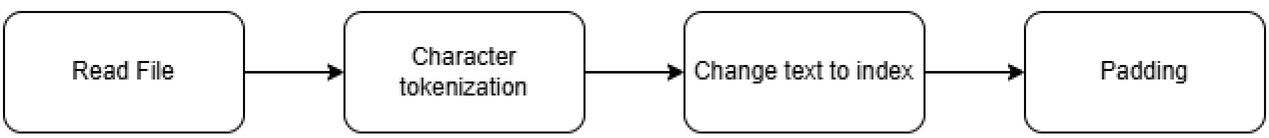

Figure 4. Tokenization

The next step after data normalization is tokenization. In this data preprocessing phase, tokenization is performed to convert text into a numerical representation easily understood by the model (Patel & Sule, 2023). The process begins by reading the input and target texts from the files, followed by character tokenization, where the text is broken down into minor units, which are characters. Each character is then assigned a numerical index based on its frequency in the dataset. Afterward, the sequences of characters are converted into numerical representations. To ensure uniformity, padding is applied to make the sequences of equal length. The model can process the text more deeply through character tokenization, identifying and correcting typos at the character level, and improving the model's ability to handle spelling variations or typing errors.

2.5. Split Data

```
X_train, X_test, y_train, y_test = train_test_split(
    combined_sequences,
    combined_labels,
    test_size=0.2,
    random_state=42,
    stratify=combined_labels)
```

Figure 5. Pseudo Code Split Data

The next step involves dividing the dataset into training and testing sets, with a random allocation of the processed data and their respective labels. 80% of the data is designated for model training, while the remaining 20% is used to assess the model's performance. This separation guarantees that the model receives enough data for training while maintaining a distinct dataset to objectively evaluate its accuracy. With this approach, the model can be pretty evaluated using unseen data, providing a more accurate picture of its ability to predict new data. Stratified splitting is employed to prevent class imbalance, ensuring that the class distributions in both the training and testing datasets are consistent with those in the original dataset. This helps prevent bias toward the majority class, ensuring that the model is trained to recognize both classes effectively and can generalize better to new, unseen data.

2.6. Long Short-Term Memory (LSTM) Model

Long Short-Term Memory (LSTM) is a type of Recurrent Neural Network (RNN) specifically designed to overcome challenges related to long-term memory in sequential data, including text (Baghoussi et al., 2024). LSTM can capture relationships and patterns in data sequences by considering previous contexts. In the context of this study, LSTM is used to effectively learn spelling error patterns and standard words, enabling it to provide more accurate automatic corrections for spelling mistakes in written works (Wiranda & Sadikin, 2019). LSTM utilizes three main gates to process information.

The Forget Gate determines which information should be discarded. The formula for calculating forget gate is as follows.

$$f_t = \sigma(W_f[h_{t-1}, x_t] + b_f) \qquad (1)$$

Description:

f_t	=	Forget Gate
x_t	=	The input at time t
h_{t-1}	=	The hidden state from the previous time
σ	=	The sigmoid activation function
W_f	=	Weight matrix
b_f	=	Bias

The Input Gate is responsible for selecting the new information to be stored. The formula for calculating input gate is as follows.

$$i_t = \sigma(W_i[h_{t-1}, x_t] + b_i) \qquad (2)$$

$$\tilde{C}_t = tanh(W_c[h_{t-1}, x_t] + b_c) \qquad (3)$$

$$\boldsymbol{C_t = f_t \,.\, C_{t-1} + i_t \,.\, \tilde{C}_t} \qquad (4)$$

Description:

i_t	=	Input Gate
x_t	=	The input at time t
h_{t-1}	=	The hidden state from the previous time
σ	=	The sigmoid activation function
W_i	=	Input gate weight
b_i	=	Input Bias
$tanh$	=	The tanh activation function
$\tilde{C}_t$	=	Candidate's new memory cell values
W_c	=	New memory cell weight
b_c	=	New memory cell Bias
C_t	=	Memory cell
f_t	=	Forget Gate
C_{t-1}	=	Memory Cells from the previous time

The Output Gate decides which section of the memory will be output. The formula for calculating output gate is as follows.

$$o_t = \sigma(W_o[h_{t-1}, x_t] + b_o) \qquad (5)$$

$$h_t = o_t \,.\, \tanh(C_t) \qquad (6)$$

Description:

o_t	=	Output Gate
x_t	=	The input at time t
h_{t-1}	=	The hidden state from the previous time
σ	=	The sigmoid activation function
W_o	=	Input gate weight
b_o	=	Input Bias
h_t	=	Hidden state
C_t	=	Cell state
tanh	=	The tanh activation function

LSTM works in automatic text correction is as follows: after tokenization, the next step involves the model using an Embedding layer to convert the numerical tokens into fixed-dimensional word vectors. These vectors represent the meaning of words in a vector space, which are processed through the three main gates of the LSTM. As a result, words with similar meanings are represented more closely to each other. In the model, the forget gate determines which information is not relevant to the context and should be discarded, the input gate identifies the new information that should be stored in memory, and the output gate selects the data that will be sent to the LSTM output.

2.7.Model Training

In the training phase, the LSTM model is trained on a processed dataset to predict whether a word in the text is a standard word or a typo. The first step performed by the model is to prepare the data used for training. This data consists of input data containing token representations and target labels to indicate whether each word in the dataset is standard or a typo. The model subsequently categorizes the words as either standard or erroneous, and is trained with the Adam optimizer to minimize the categorical cross-entropy loss function, which is commonly used for multi-class classification tasks. Accuracy serves as the main evaluation metric to assess the model's performance throughout the training process. Throughout the training process, validation data is utilized to minimize the risk of overfitting, and the outcomes of each epoch are tracked to ensure the model is learning properly without overfitting or underfitting.

2.8.Model Evaluation

The model evaluation phase utilizes a confusion matrix to evaluate the performance of the model in automatically and accurately correcting text. The confusion matrix is an essential tool for evaluating classification performance, offering a clear view of how the model makes predictions and identifies areas where errors happen (Riehl et al., 2023). The confusion matrix consists of four main elements that describe the relationship between predicted outcomes and actual labels: True Positive (TP), False Positive (FP), True Negative (TN), and False Negative (FN):

a. True Positif (TP): The number of positive data points and correctly predicted as positive.
b. False Positif (FP): The number of data points that are actually negative but incorrectly predicted as positive.
c. False Negatif (FN): The number of data points that are actually positive but incorrectly predicted as negative.
d. True Negatif (TN): The number of negative data points and correctly predicted as negative

The confusion matrix helps analyze the strengths and weaknesses of a model in text correction and identify areas that need improvement, such as reducing the number of False Positives or increasing the detection of False Negatives. The confusion matrix can also be used to derive several evaluation metrics, including accuracy, precision, recall, and F1-score, offering a more comprehensive insight into the model's performance across various scenarios (Sathyanarayanan & Tantri, 2024). Accuracy is a testing approach that evaluates how closely the model's predictions align with the actual values. By calculating the number of data points that are correctly classified, we can estimate the accuracy level of the predictions. The formula for calculating accuracy is as follows.

$$Accuracy = \frac{TP+TN}{TP+FP+FN+TN} \qquad (7)$$

Precision is a testing technique that measures the proportion of relevant information identified by the system in relation to the total information retrieved by the system. Precision reflects the likelihood that cases predicted as positive are actually in the positive category. The formula for calculating precision is as follows.

$$Precision = \frac{TP}{TP+FP} \qquad (8)$$

Recall is a testing method used to assess the probability of accurately identifying positive cases within the positive category. The formula for calculating recall is as follows.

$$Recall = \frac{TP}{TP+FN} \tag{9}$$

3. RESULTS AND DISCUSSION

3.1 Preprocessing Results

At this stage, the researcher will present the results of the experiments to measure the model's performance in automatic text correction. Before starting the model training process, the first step was to collect data. The researcher has gathered 45,698 standard words from the KBBI platform. An example is shown in **Table 1**.

Table 1. Standard Word

Index	Standard Word
0	abad
1	berabad
2	makan
3	pengabdian
4	keabadian
5	malam
6	pulang
7	pergi
8	laptop
9	pintu

Subsequently, the researcher normalized the standard words before introducing variations of spelling errors or typos. This normalization process ensures that the data used for model training is clean and consistent. The first step in normalization involves converting all letters to lowercase to eliminate inconsistencies caused by differences in capitalization. Next, duplicate words, which may have resulted from input errors, are removed to maintain data integrity. Additionally, irrelevant symbols or characters that could interfere with the model's ability to recognize actual patterns are removed. Equally important, numbers unrelated to the context are also eliminated, ensuring that the analysis remains focused on words. After the normalization stage is completed, the researcher then proceeds with data augmentation on the collected dataset. Data augmentation is carried out to increase the dataset's diversity by introducing variations of spelling mistakes or typos for each standard word. These variations include letter substitution errors, letter deletion, letter insertion, and letter position transposition within words. Therefore, the augmented data is expected to represent the various types of errors that may occur in real-world scenarios, enabling the model to be more robust and make more accurate corrections. This is illustrated in Table 2.

Table 2. Typo Word

Index	Typo Word
0	erabad, brabad, berabad
1	megabadi, engabadi, mngabadi
2	mngabaikan, engabaikan, megabaikan
3	pngabai, pegabai, engabai
4	mngabdikan, engabdikan, megabdikan
5	pnganggur, peganggur, enganggur
6	eranggapan, beanggapan, branggapan
7	engamputasian, pegamputasian, pngamputasian
8	engampunan, pegampunan, pngampunan
9	mngampuni, engampuni, megampuni

3.2 Model Training Results

After applying data augmentation, the researcher trains the LSTM model with an enriched dataset containing multiple variations of typing errors for each standard word. The training process involves feeding the processed data into the model so that it can learn the patterns of spelling errors and the relationships between words in the text. During training, the LSTM model will attempt to optimize its weights through a learning algorithm. The model training process can be seen in Table 3.

Table 3. Model Training

Epochs	Accuracy	Loss	Accuracy Validation	Loss Validation
5	0,92	0,15	0,93	0,15
10	0,93	0,15	0,93	0,14
15	0,94	0,12	0,94	0,13
20	0,96	0,08	0,94	0,15
25	0,96	0,08	0,93	0,15

The results of model training with different epoch counts demonstrate a notable increase in accuracy and a reduction in loss as the number of epochs grows. At epoch 5, the training accuracy reached 92% with a loss of 0.15, while the validation accuracy was 93% with a loss of 0.15. At epoch 10, the training accuracy increased to 93% with a loss of 0.15, followed by a stable validation accuracy of 93%, accompanied by a decrease in loss to 0.14. Subsequently, at epoch 15, the training accuracy further increased to 94% with a loss of 0.12, while the validation accuracy remained at 94%, with a slight increase in loss to 0.13. At epoch 20, although the training accuracy remained high at 96% with a loss of 0.08, the validation accuracy slightly decreased to 94%, while the validation loss remained stable at 0.15. At epoch 25, the training accuracy stayed stable at 96% with a loss of 0.08, and the validation accuracy slightly dropped to 93%, followed by a validation loss that remained at 0.15. In general, the model showed a notable increase in training accuracy, although the validation accuracy varied after peaking at epoch 15. Based on the training results, it can be observed that the model successfully trained the data, as evidenced by the high accuracy levels and good validation results.

Further evaluation shows that the model demonstrates stable performance throughout the training process. The graphs in Figure 6 show the progression of accuracy and loss during training.

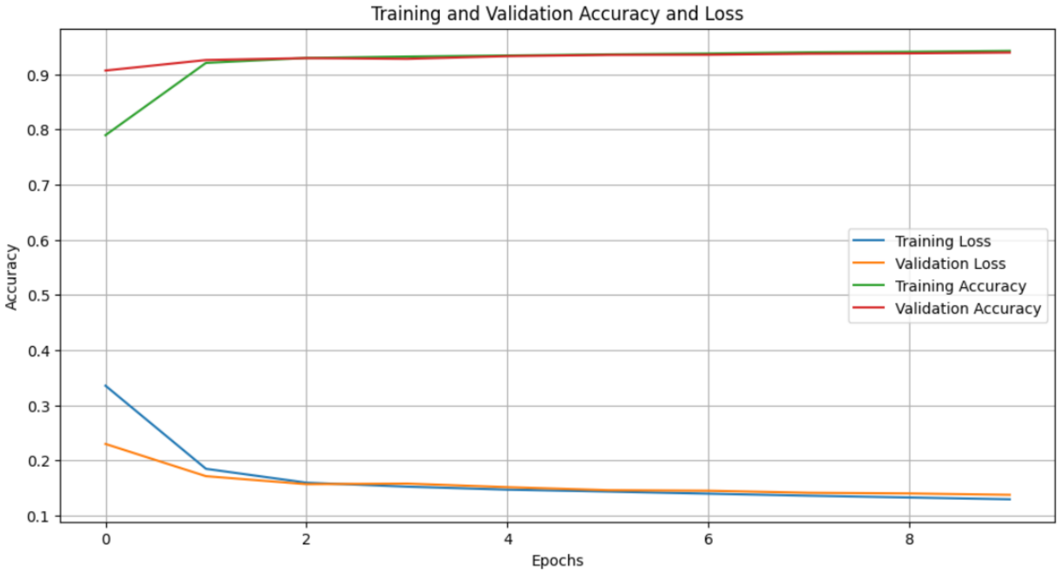

Figure 6. Training and Validation (Loss and Accuracy)

Figure 6 shows a decrease in the loss value. The graph illustrates a reduction in loss for the model using 5 epochs, indicating that the model is improving in minimizing prediction errors and producing more accurate correction outputs.

Figure 6 illustrates the improvement in the model's accuracy. As shown in the figure, both the training accuracy and validation accuracy of the model continue to increase, indicating that the model is becoming more effective in training its data. After training the model, the researcher conducted testing on the trained model, as shown in **Table 4**.

Table 4. Testing

Typo Words	Model Prediction	Standard Word	Status
ikn	ikan	ikan	True
makn	makan	makan	True
mkan	akan	makan	False
peakut	penakut	penakut	True
selmat	selmat	selamat	False
adpun	adapun	adapun	True
menghrup	menghirup	menghirup	True
membca	membaca	membaca	True
kelompk	kelompok	kelompok	True
adpaun	adpaun	adapun	False

The results show that the system can predict words effectively, although there are still some errors in its text correction. One type of error that can occur is when the input word is split or scrambled, making it difficult for the system to recognize and correct it properly. For example, the word "adapun" might be divided into "adaupn" or even have its letters scrambled, causing the system to fail to detect and accurately correct the error. Furthermore, when two letters are deleted from a word, the system still struggles to detect the intended correction as expected. However, if only one letter is removed, the system is more capable of identifying and suggesting the most appropriate correction. Additionally, inserting extra letters also presents a challenge for the system, making it more difficult to detect and correct errors effectively. Next, the trained model's performance is evaluated to ensure its accuracy and effectiveness in text correction. This evaluation is performed using a confusion matrix, an important tool for assessing how well the model predicts correctly or incorrectly and for calculating various evaluation metrics such as precision, recall, and F1-score. These results can be seen in Table 5.

Table 5. Classification Report

	Precision	Recall	F1-score	Support
Typo	0,95	0,96	0,96	27403
Baku	0,89	0,92	0,86	9156
Accuracy			0,93	36559
Macro avg	0,92	0,90	0,91	36559
Weighted avg	0,93	0,93	0,93	36559

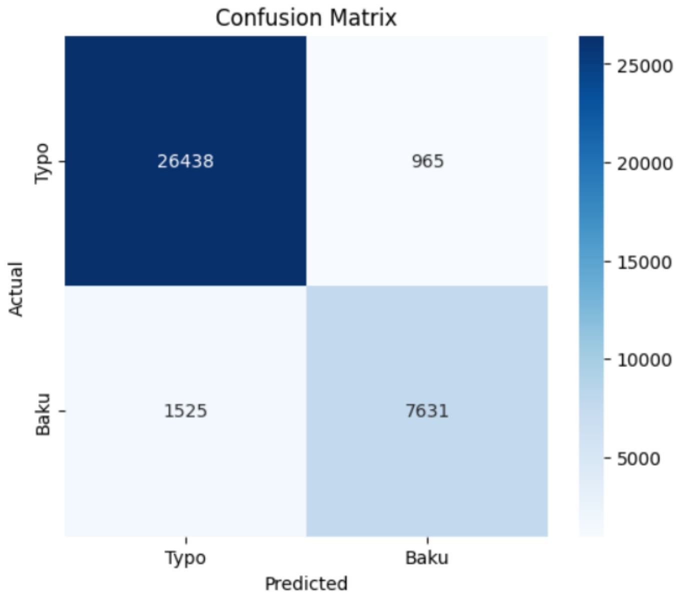

Figure 7. Confusion Matrix

The results from the confusion matrix show that the True Positive was detected 7,631 times, where standard words were correctly identified as standard words. The True Negative was detected 26,438 times, where the model correctly identified words that should be standard but incorrectly detected as typos. Furthermore, the False Positive was detected 965 times, where typos were correctly identified as typos, and the False Negative was detected 1,525 times, where typos were incorrectly identified as standard words. These results can be seen in **Figure 7**.

3.3 Discussion

In its implementation, the LSTM-based text correction system successfully detects and corrects spelling errors in Indonesian scientific writing with a high degree of accuracy. The model demonstrates robust performance, achieving a validation accuracy of 93%, along with consistently high precision, recall, and F1-score metrics. This indicates that the system's training and validation processes were effectively executed, resulting in a reliable model capable of handling various typographical errors. However, certain challenges remain, particularly in cases involving scrambled letters, multiple missing characters, or excessive letter insertions. These limitations suggest opportunities for further improvement in future iterations of the system to enhance its ability to address more complex error patterns. The findings of this study are consistent with those of Kusuma & Ratnasari (2023), who compared the performance of LSTM, Peter Norvig, and N-Gram algorithms in spelling correction, highlighting the superior accuracy of the Peter Norvig approach while acknowledging LSTM's effectiveness in sequential data handling. Similarly, Zaky & Romadhony (2019) reported an LSTM accuracy of 83.76% in handling addition, deletion, substitution, and transposition errors, while Yang et al. (2020) emphasized the importance of tailored preprocessing and augmentation techniques for language-specific datasets. Despite the promising results, optimizing the LSTM model for complex spelling errors remains a challenge. Future research could integrate Transformer-based models, such as BERT, to improve contextual understanding and correction accuracy. Additionally, expanding the dataset with more diverse and realistic error patterns could further enhance the model's performance across different writing scenarios. Overall, the LSTM-based text correction system represents a significant advancement in improving the clarity and precision of Indonesian scientific writing, offering great potential as a reliable tool for academic and professional text correction tasks.

4. CONCLUSION

This research successfully developed an automatic text correction system using a Natural Language Processing (NLP) approach with a Long Short-Term Memory (LSTM) model designed to detect and correct spelling errors in academic papers. The system underwent several stages, including data collection, normalization, model training, and evaluation using a confusion matrix. The training results showed a training accuracy of 93% and a validation accuracy of 93%, indicating that the model performed well in handling various typing error patterns. The inclusion of typo variations through data augmentation significantly enhanced the model's ability to identify and correct different types of errors. However, some inconsistencies were still observed during testing, such as difficulties in recognizing and correcting split or scrambled words, challenges in making corrections when two letters were deleted compared to better performance with single-letter deletions, and decreased accuracy in handling extra letter insertions. For future development, it is recommended to expand the corpus with more complex and diverse error variations and integrate the LSTM model with transformer-based algorithms like BERT to improve contextual understanding, enhance text correction accuracy, and optimize the overall model performance.

REFERENCES

Adawiyah, R. (2023). Analisis kesalahan penulisan paragraf bahasa Inggris oleh mahasiswa non-jurusan bahasa Inggris. *Innovative: Journal of Social Science Research*, *3*(6), 7308–7320.

Amien, M. (2023). Sejarah dan perkembangan teknik Natural Language Processing (NLP) bahasa Indonesia: Tinjauan tentang sejarah, perkembangan teknologi, dan aplikasi NLP dalam bahasa Indonesia. *Research Gate*. https://www.researchgate.net/publication/369855102_Sejarah_dan_Perkembangan_Teknik_Natural_Language_Processing_NLP_Bahasa_Indonesia_Tinjauan_tentang_sejarah_perkembangan_teknologi_dan_aplikasi_NLP_dalam_bahasa_Indonesia

Baghoussi, Y., Soares, C., & Mendes-Moreira, J. (2024). Corrector LSTM: Built-in training data correction for improved time-series forecasting. *Neural Computing and Applications*, *36*(26), 16213–16231. https://doi.org/10.1007/s00521-024-09962-x

Dewi, N. C., & Qoiriah, A. (2021). Implementasi algoritma jaro-winkler distance dan N-Gram untuk deteksi dan prediksi perbaikan kesalahan penulisan kata bahasa Indonesia pada karya tulis ilmiah mahasiswa. *Journal of Informatics and Computer Science*, *2*(03), 169–177. https://doi.org/10.26740/jinacs.v2n03.p169-177

Herawati, I., Kanzunnudin, M., & Wiranti, D. A. (2022). Analisis kesalahan ejaan dalam penulisan karangan deskripsi siswa kelas IV SD 04 Besito Kudus. *Jurnal Prasasti Ilmu*, *2*(3), 128–132. https://doi.org/10.24176/jpi.v2i3.8643

Juniarti, Y. (2019). Pentingnya keterampilan menulis akademik di perguruan tinggi. *Prosiding Sembadra Universitas Sriwijaya*, *2*(1), 185–189.

Khaidir, J., Erlinawati, Sriani, Y., & Hidayat, A. (2023). *Teknik penulisan karya ilmiah* (N. Saputra (ed.); Vol. 1, Issue February). Yayasan Penerbit Muhammad Zaini. https://www.google.co.id/books/edition/pengantar_teknik_penulisan_karya_ilmiah/nx7eeaaaqbaj?hl=id&gbpv=0

Kusuma, A. T., & Ratnasari, C. I. (2023). Comparison of spell correction in bahasa Indonesia: Peter norvig, LSTM, and N-Gram. *JIKO (Jurnal Informatika Dan Komputer)*, *6*(3), 214–220. https://doi.org/10.33387/jiko.v6i3.7072

Marlina, Y. I. (2019). *Bentuk kesalahan berbahasa ruang publik: kajian struktural bahasa* [Thesis, Universitas Muhammadiyah Surakarta]. https://eprints.ums.ac.id/76214/1/NASKAH PUBLIKASI

Patel, B. M., & Sule, M. (2023). Tokenization techniques in NLP: A comprehensive review. *International Journal of Advance Research and Innovative Ideas in Education*, *9*(1), 1873–1892. https://ijariie.com/adminuploadpdf/tokenization_techniques_in_nlp_a_comprehensive_review_ijariie22082.pdf

Putri, R. R., & Cahyono, N. (2024). Analisis sentimen komentar masyarakat terhadap pelayanan publik pemerintah DKI Jakarta dengan algoritma super vector machine and naive bayes. *JATI (Jurnal Mahasiswa Teknik Informatika)*, *8*(2), 2363–2371. https://doi.org/10.36040/jati.v8i2.9472

Rayhan, A., Kinzler, R., & Rayhan, R. (2023). Natural language processing: Transforming how machines understand human language. *Researchgate*. https://doi.org/10.13140/RG.2.2.34900.99200

Riehl, K., Neunteufel, M., & Hemberg, M. (2023). Hierarchical confusion matrix for classification performance evaluation. *Journal of the Royal Statistical Society. Series C: Applied Statistics*, *72*(5), 1394–1412. https://doi.org/10.1093/jrsssc/qlad057

Rosmiati, A. (2017). Dasar-dasar penulisan karya ilmiah., *ISI Press*. http://repository.isi-ska.ac.id/1395/3/Dasar-Dasar Penulisan Ilmiah.pdf

Rustanti, H. D. (2024). Analisis kesalahan penggunaan ejaan bahasa indonesia pada karya ilmiah siswa kelas XI SMA Negeri 86 Jakarta Tahun Pelajaran 2021/2022. In *UIN*. https://repository.uinjkt.ac.id/dspace/bitstream/123456789/77844/1/HANIFAH DWI RUSTANTI11180130000023.pdf

Sathyanarayanan, S., & Tantri, B. R. (2024). Confusion matrix-based performance evaluation metrics. *Afr. J. Biomed. Res.*, *27*(4), 4023–4031. https://doi.org/10.53555/AJBR.v27i4S.4345

Siregar, S., Hasibuan, N. S., & Harahap, E. M. (2023). Pengaruh penggunaan teknik koreksi secara langsung pada keterampilan menulis puisi siswa kelas X di SMA Negeri I Siabu. *Linguistik: Jurnal Bahasa Dan Sastra*, *8*(3), 449–513. http://jurnal.um-tapsel.ac.id/index.php/Linguistik/article/view/12837/pdf

Suhendar, A., Sugiarti, D. H., & Rosalina, S. (2023). Analisis kesalahan penulisan judul pada berita online Karawangpost.com dan Purwakartanews.com. *Jurnal Onoma: Pendidikan, Bahasa, Dan Sastra*, *9*(1), 113–124. https://doi.org/10.30605/onoma.v9i1.2141

Suprihatma. (2024). Analisis penggunaan bahasa Indonesia dalam jurnalistik pada media massa online. *Journal on Education*, *6*(2), 11011–11018. https://doi.org/10.31004/joe.v6i2.4892

Utama, F. P., Nurhadi, R. M. H., Fitria, D., & Ramadhan, M. P. (2021). Studi perbandingan implementasi string matching dengan metode sequential searching dan kondisi like pada pencarian judul skripsi. *Jurnal Rekursif*, *9*(1), 43–47. https://doi.org/10.33369/rekursif.v9i1.14315

Wiranda, L., & Sadikin, M. (2019). Penerapan long short term memory pada data time series untuk memprediksi penjualan produk PT. Metiska Farma. *Jurnal Nasional Pendidikan Teknik Informatika (JANAPATI)*, *8*(3), 184–196.

Susanti, W., Wulandari, W., Hasanah, U., Aprindah, & Wahyuni,. (2022). Analisis kesalahan berbahasa pada berita dalam media surat kabar Kompas.com. *KASTRAL: Kajian Sastra Nusantara Linggau*, *2*(2), 1–8. https://doi.org/10.55526/kastaral.v2i2.277

Yang, Z., Zeng, H., & Li, H. (2020). Chinese text error correction method based on prefix tree merging. *IEEE 3rd International Conference on Automation, Electronics and Electrical Engineering,* 272–276. https://doi.org/10.1109/AUTEEE50969.2020.9315643

Yulianizar, R., & Waliah, S. Z. (2022). Analisis kesalahaan ejaan terhadap teks berita “Bikin gagal ginjal, etilen glikol di obat sirup ternyata 'familiar’ di mesin” pada media online Detikoto”. *Sinar Dunia: Jurnal Riset Sosial Humaniora dan Ilmu Pendidikan*, *1*(4), 62–73. https://doi.org/10.58192/sidu.v1i4.225

Zaky, D., & Romadhony, A. (2019). An LSTM-based spell checker for Indonesian text. *Proceedings - 2019 International Conference on Advanced Informatics: Concepts, Theory, and Applications,* 1–6. https://doi.org/10.1109/ICAICTA.2019.8904218