Accredited SINTA 2 Ranking
Decree of the Director General of Higher Education, Research, and Technology, No. 158/E/KPT/2021
Validity period from Volume 5 Number 2 of 2021 to Volume 10 Number 1 of 2026

Published online at: **http://jurnal.iaii.or.id**

JURNAL RESTI
(Rekayasa Sistem dan Teknologi Informasi)
Vol. 9 No. 2 (2025) 327 - 333 e-ISSN: 2580-0760

Sentiment Analysis of ChatGPT on Indonesian Text using Hybrid CNN and Bi-LSTM

Vincentius Riandaru Prasetyo[1]*, Mohammad Farid Naufal[2], Kevin Wijaya[3]
[1,2,3]Department of Informatics, Faculty of Engineering, University of Surabaya, Surabaya, Indonesia
[1]vincent@staff.ubaya.ac.id, [2]faridnaufal@staff.ubaya.ac.id, [3]s160420136@student.ubaya.ac.id

Abstract

This study explores sentiment analysis on Indonesian text using a hybrid deep learning approach that combines Convolutional Neural Networks (CNN) and Bidirectional Long Short-Term Memory (Bi-LSTM). Due to the complex linguistic structure of the Indonesian language, sentiment classification remains challenging, necessitating advanced methods to capture both local patterns and sequential dependencies. The primary objective of this research is to improve sentiment classification accuracy by leveraging a hybrid model that integrates CNN for feature extraction and Bi-LSTM for contextual understanding. The dataset consists of 800 manually labeled samples collected from social media platforms, preprocessed using case folding, stop word removal, and lemmatization. Word embeddings are generated using the Word2Vec CBOW model, and the classification model is trained using a hybrid architecture. The best performance was achieved with 32 Bi-LSTM units, a dropout rate of 0.5, and L2 regularization, which was evaluated using Stratified K-Fold cross-validation. Experimental results demonstrate that the hybrid model outperforms conventional deep learning approaches, achieving 95.24% accuracy, 95.09% precision, 95.15% recall, and 95.99% F1 score. These findings highlight the effectiveness of hybrid architectures in sentiment analysis for low-resource languages. Future work may explore larger datasets or transfer learning to enhance generalizability.

Keywords: sentiment analysis; CNN; Bi-LSTM; hybrid model

How to Cite: V. R. Prasetyo, M. F. Naufal, and K. Wijaya, "Sentiment Analysis of ChatGPT on Indonesian Text using Hybrid CNN and Bi-LSTM", *J. RESTI (Rekayasa Sist. Teknol. Inf.)* , vol. 9, no. 2, pp. 327 - 333, Apr. 2025.
DOI: https://doi.org/10.29207/resti.v9i2.6334

1. Introduction

ChatGPT is an artificial intelligence (AI) model developed by OpenAI. This model uses advanced machine learning technology, namely transformers, to understand and generate text in human language. ChatGPT is trained using various text data from the internet, so it can answer questions, provide explanations, discuss, or even create creative content such as stories, poems, or articles [1]. ChatGPT has several advantages that make it very useful in various contexts. It can communicate in many languages and understand and respond to conversations in a relevant context. In addition, ChatGPT can generate creative text in various forms, such as essays, stories, or programming code, which makes it ideal for writing or developing ideas. Users can also personalize the conversation style, whether formal or casual, according to their needs. ChatGPT can also complete complex tasks, such as explaining technical concepts or providing advice, and can be accessed through various platforms flexibly and securely. Its advantages in scalability and ability to support various types of applications, from education to customer service, make it a convenient and efficient tool [2].

While ChatGPT in education offers various benefits, some threats and challenges must be considered. One is the potential for misuse for plagiarism, where students can use ChatGPT to write assignments without understanding the material or developing their writing skills. Over-reliance on this technology can also reduce students' ability to think critically and solve problems independently. In addition, ChatGPT does not always provide accurate information, which can spread misinformation if used without verification and hinder the development of research and writing skills that should be trained in the learning process [3]. The gap in access to technology is also an issue, as not all students have the same tools to utilize this AI, creating inequities in learning. Overuse of ChatGPT can reduce student engagement in class discussions and lead to a shallower

Received: 28-01-2025 | Accepted: 06-04-2025 | Published Online: 11-04-2025

understanding of the material, especially in more complex topics. From an ethical perspective, using AI in education raises questions about the authenticity of assignments and fair assessments and how to assess students' abilities using this technology. Finally, reliance on ChatGPT can reduce social interaction and communication between students and instructors, which is essential for developing social and discussion skills. Thus, although ChatGPT offers convenience, it is important to use this technology wisely so as not to reduce the quality of learning and increase the student gap [4].

Deep learning techniques have been investigated several times for sentiment analysis. While Bidirectional Long Short-Term Memory (Bi-LSTM) networks shine in capturing sequential and contextual linkages, Convolutional Neural Networks (CNNs) have proven rather strong in extracting local text properties. Combining CNN with Bi-LSTM has shown potential in using the advantages of both models in hybrid techniques [5]. On the IMDB dataset, the combination had 93.3% accuracy, where CNN structure helps extract local aspects of the text. Bi-LSTM completes the context information interaction by allocating weight and resources to the text material of many, which helps to increase performance [6].

Another study using the Arabic dataset showed that hybrid CNN and Bi-LSTM performed well for binary and multiclass classification with accuracy of 98.47% and 98.92%, respectively [7]. Another study using a tweets dataset showed that the hybrid CNN and Bi-LSTM had an accuracy of 82%, better than the implementation of Bi-LSTM alone, with an accuracy of 76% [8]. In addition, using airline quality and Twitter airline sentiment datasets, the combination of CNN and Bi-LSTM also produces promising performance with an accuracy of 91.3% [9].

Specific research related to sentiment analysis from ChatGPT has been conducted using a dataset on Twitter and comparing the accuracy of 2 machine learning methods, namely Support Vector Machine (SVM) and Naïve Bayes. From the evaluation results, the two methods were not better than the hybrid CNN and Bi-LSTM methods, with the best accuracy produced being 59% and 47% [10]. Another study using a Twitter dataset for sentiment analysis of ChatGPT was conducted by combining C4.5 and Naïve Bayes methods. The C4.5 algorithm was employed to discern Twitter usage trends, whereas the Naïve Bayes algorithm was utilized to categorize the predominant forms of interactions between Twitter users and ChatGPT. The amalgamation of the two techniques yielded an accuracy of 77.33% [11].

2. Research Methods

The processes that occur in this research consist of several stages, including dataset collection, preprocessing, word embedding, model training, model evaluation, and implementation. The flowchart of the stages in this study can be seen in Figure 1. The process begins with collecting datasets through crawling techniques from social media, followed by manual labeling into positive, negative, and neutral categories. The data then undergoes preprocessing, including case folding, removal of links and special characters, stop word removal, and lemmatization to improve text quality. Next, word embedding using the Continuous Bag of Words (CBOW) method from Word2Vec is applied to represent text in vector form. A hybrid CNN and Bi-LSTM model is used for sentiment classification, with CNN extracting local features through convolutional and max-pooling layers. At the same time, Bi-LSTM captures sequential relationships in text before being processed by a dense layer with SoftMax activation. The model is trained using Stratified K-Fold cross-validation, and hyperparameter tuning is performed by testing variations in the number of Bi-LSTM units, dropout, and regularizer (L1/L2). The model evaluation uses accuracy, precision, recall, and F1-score metrics to measure model performance in accurately classifying sentiment. This figure provides a comprehensive overview of the research process, from data collection to model evaluation and the application of hybrid architecture in Indonesian language sentiment analysis.

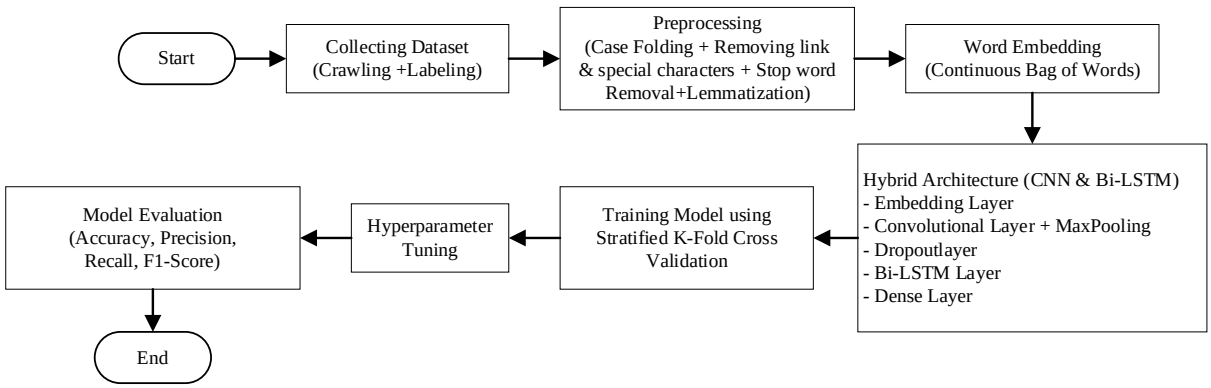

Figure 1. Research Steps

2.1 Collecting Dataset

In this study, datasets were collected using crawling techniques. Crawling is an automatic data extraction technique from websites that can be stored and analyzed [12]. Indonesian language text opinion data was taken from YouTube, Instagram, and X platforms. The three platforms were chosen because they are included in the top 10 social media platforms that are widely used by Indonesian people today [13].

Crawling is done by utilizing the selenium library. Selenium is one of the popular libraries used for web scraping or crawling, especially when the website uses a lot of JavaScript to load content [14]. An expert, Anis Sumiati, S.Pd., a Surabaya Cambridge School High School teacher, will manually label the collected dataset. The dataset is labeled into three classes: positive, negative, and neutral, where each class has a total of 392, 243, and 165 data. Table 1 shows some examples of datasets and their labels.

Table 1. Example Dataset and Its Sentiment Labels

Comment Data	Sentiment Label
Banyaak banget 😊. Kalau lagi baca buku atau artikel penelitian trus ada kalimat yg nggak paham biasanya aku pake chatGPT buat bantu jelasin	Positive
Bagus buat bikin tugas, tapi lama2 menurunkan intelligent siswa. Terbukti, para siswa kalau dikasih tugas paper cepet selesai tapi begitu diajak diskusi isinya Cuma bisa tolah toleh bego	Negative
Asli enak banget pakai ChatGPT buat bantu rephrase in academic term dalam nulis jurnal. Secara kapan coba gw nulis jurnal, jadi confusing bener	Positive
Penggunaan AI sejauh ini yg tak approve ya buat riset, ChatGPT sama Bard bener2 life changing for my life. Walaupun suwi2 aku wedi sisan, kemampuan literasi ku bakal menurun hhh	Neutral

2.2 Preprocessing

After the dataset is collected and labeled, preprocessing is an important stage in sentiment analysis. Preprocessing involves several processes to format text that are useful for improving the performance of the classification model built [15]. This study carried out several processes: case folding, cleaning, stop word removal, and lemmatization.

The first process is case folding, which aims to transform all character variations in the text into uniform and consistent ones so that the same word is not represented as two different vectors. This process is important because deep learning models are case-sensitive. Deep Learning treats uppercase and lowercase letters as different entities [16].

The following preprocessing removes links, mentions, punctuation, hashtags, characters, and excess spaces and changes symbols to their actual meaning. This process is important to avoid elements that do not represent the sentiment of an opinion, so it needs to be cleaned so as not to affect the classification results [16].

The third preprocessing is stop word removal, where unimportant words in the classification process will be removed. The categories of word types included as stop words are prepositions, conjunctions, and pronouns [17]. The last preprocessing is lemmatization, where this process is like stemming because both reduce word variants into basic word forms, but the process is different. Stemming changes words into basic word forms by removing prefixes and suffixes, while lemmatization changes basic words or lemmas by considering the grammar rules in the dictionary. Table 2 shows an example of the preprocessing results of the dataset.

Table 2. Example of Preprocessing Results

Original Text	Preprocessing Result
Banyaak banget 😊. Kalau lagi baca buku atau artikel penelitian trus ada kalimat yg nggak paham biasanya aku pake chatGPT buat bantu jelasin	*banyaak banget baca buku artikel teliti trus kalimat yg nggak paham pake chatgpt buat bantu jelasin*
Bagus buat bikin tugas, tapi lama2 menurunkan intelligent siswa. Terbukti, para siswa kalau dikasih tugas paper cepet selesai tapi begitu diajak diskusi isinya Cuma bisa tolah toleh bego	*bagus buat bikin tugas tapi turun intelligent siswa bukti siswa dikasih tugas paper cepet selesai tapi diajak diskusi isinya cuma bisa tolah toleh bego*
Asli enak banget pakai ChatGPT buat bantu rephrase in academic term dalam nulis jurnal. Secara kapan coba gw nulis jurnal, jadi confusing bener	*asli enak banget pakai chatgpt buat bantu rephrase in academic term nulis jurnal kapan coba gw nulis jurnal confusing bener*
Penggunaan AI sejauh ini yg tak approve ya buat riset, ChatGPT sama Bard bener2 life changing for my life. Walaupun suwi2 aku wedi sisan, kemampuan literasi ku bakal menurun hhh	*guna ai yg tak approve ya buat riset chatgpt sama bard bener life changing for my life suwi wedi sisan mampu literasi ku turun hhh*

2.3. Word Embedding

The next stage after preprocessing is word embedding. The word embedding model used in this study is Word2Vec. The purpose of Word2Vec is to capture the similarity between words in the text using vector representation. Word2Vec has two algorithms: Continuous Bag of Words (CBOW) and Skip-gram [18]. The method applied in this work is CBOW. This algorithm predicts one target word with one context word as input. CBOW is chosen since it fits big datasets [18] and offers the advantage of fast data processing. Figures 2 show CBOW's architecture.

The CBOW approach starts with figuring the window size—e.g., 2—where for every target word (center word), we employ the two words before and two words following as context words [18]. Forming a context-target pair, as described in Table 3, comes next; an example sentence might be "*ChatGPT adalah model AI yang sangat populer*." After that, the words in the sentence are represented as one-hot encoding, as shown in Table 4.

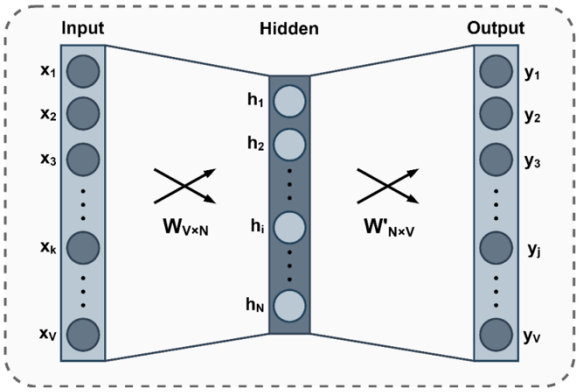

Figure 2. CBOW Architecture

Table 3. Example of Context-Target Pair

Context Words	Target Word
["ChatGPT", "adalah"]	*"model"*
["adalah", "model", "AI"]	*"yang"*
["model", "AI", "yang"]	*"sangat"*
["AI", "yang", "sangat"]	*"populer"*

Table 4. One-hot Encoding Representation of Each Word

Word	One-hot encoding
ChatGPT	[1, 0, 0, 0, 0, 0, 0]
adalah	[0, 1, 0, 0, 0, 0, 0]
model	[0, 0, 1, 0, 0, 0, 0]
AI	[0, 0, 0, 1, 0, 0, 0]
yang	[0, 0, 0, 0, 1, 0, 0]
sangat	[0, 0, 0, 0, 0, 1, 0]
populer	[0, 0, 0, 0, 0, 0, 1]

The next step is the process of forming the CBOW model [18], where in this example, the target word is "*model*," and the context words are ["*ChatGPT*," "*adalah*"]. Next, we calculate the average vector of the context word so that the new vector is [0.5,0.5,0,0,0,0,0]. The calculation results will be forwarded to the hidden layer to map the one-hot vector to a smaller dimensional embedding space [18], for example, from dimension 7 (number of vocabulary) to dimension 3, so that the new vector is [0.3,0.7,0.2].

The following process involves forwarding the embedding from the hidden layer to the output layer, where the probability calculation for all words in the vocabulary occurs [18]. The results of the probability calculation for all words in the vocabulary can be seen in Table 5.

Table 5 The Results of The Probability Calculation

Word	Probability
ChatGPT	0.05
adalah	0.1
model	**0.65**
AI	0.1
yang	0.05
sangat	0.03
populer	0.02

The CBOW model predicts the term with the highest likelihood, "*model*," based on the computation results in Table 5 as the target word. Should the target word prediction be erroneous, backpropagation will update the weights in the hidden layer and output layer, improving the model's ability to predict target words depending on the context [18].

2.4 Training Model

Training the dataset with hybrid CNN and Bi-LSTM methods comes next, following the word embedding process to generate a precise classification model. This hybrid enables the model to manage text more holistically, from the local level (n-grams) to the global (word relationships in the framework of the whole text). Bi-LSTM uses local-level significant aspects to grasp the link between words in a larger context, while CNN can detect these qualities. In addition, by using CNN first for feature extraction, the length of the data sequence can be reduced, thus speeding up the process in Bi-LSTM. This combination also often produces higher accuracy than using only CNN or LSTM alone, especially on text datasets with complex patterns [5]–[9].

This study uses a hybrid CNN and Bi-LSTM architecture for sentiment analysis of Indonesian language text. This combination utilizes the advantages of CNN in extracting local features from text and Bi-LSTM's ability to understand sequential relationships between words in a global context [6]. In the first stage, the text that has gone through preprocessing is converted into a vector representation using Word2Vec with the CBOW method. This vector representation becomes input for the Convolutional Neural Network (CNN) layer, which extracts local features from text data. CNN uses a 3x3 filter (kernel) with a ReLU activation function that increases non-linearity in feature extraction. After going through the convolution process, the results are processed through Max-Pooling to reduce the dimensions of the resulting features so that computational efficiency increases, and the risk of overfitting is reduced [5]. Furthermore, the results from CNN are sent to the Bi-LSTM layer consisting of 32 units with two directions (forward and backwards), allowing the model to understand the contextual relationship before and after the words in the text. Finally, the results from Bi-LSTM are processed through a Dense Layer with a Softmax activation function to classify text sentiment into three classes: positive, negative, and neutral [8]. The hybrid CNN and Bi-LSTM architecture used in this study can be seen in Figure 3.

Based on Figure 3, the CNN architecture has 64 filters with a kernel size 3x3, allowing the model to capture important patterns in the text. The activation function used is ReLU, which helps avoid the vanishing gradient problem and improves the model's ability to extract non-linear features. Max-Pooling with a pool size of 2x2 is applied after the convolution layer to avoid overfitting, and a dropout of 0.5 is used in certain layers during training. Meanwhile, in the Bi-LSTM layer, 32 units are used with a dropout of 0.5 to maintain model generalization. In addition, the L2 regularizer is applied to prevent the model from becoming too complex and

reduce the risk of overfitting. The Softmax activation function in the output layer is used to classify text into three sentiment classes based on the highest probability. In this study, parameter selection is based on hyperparameter tuning experiments to obtain the best performance of hybrid CNN and Bi-LSTM models. Tuning was performed on units, dropout, and the regularizer. Table 6 shows the values used for tuning the three parameters. Tuning is done by testing various combinations of these three parameters.

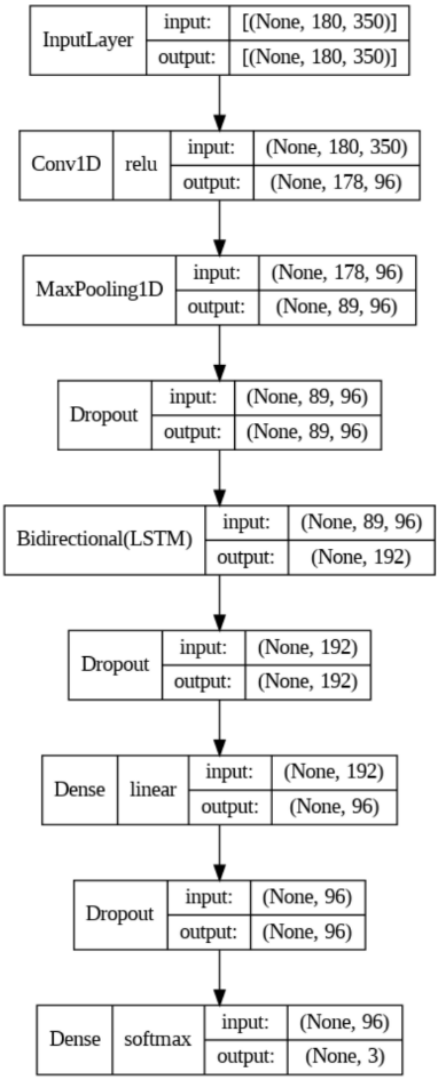

Figure 3. Hybrid Architecture of CNN and Bi-LSTM

Table 6. Hyperparameter Tuning Values

Parameters	Value
Units	32, 64, and 96
Dropout	0.4 and 0.5
Regularizer	L1 and L2

2.5 Model Evaluation

The sentiment analysis model's performance will be evaluated using four metrics: accuracy, precision, recall, and f1-score. The formulas for these four metrics can be seen in Equations 1 - 4, where *TP* is true positive, *TN* is true negative, *FP* is false positive, and *FN* is false negative [19].

$$Accuracy = \frac{TP}{TP+TN+FP+FN} \quad (1)$$

$$Precision = \frac{TP}{TP+FP} \quad (2)$$

$$Recall = \frac{TP}{TP+FN} \quad (3)$$

$$F1 - Score = \frac{2\ x\ Precision\ x\ Recall}{Precision + Recall} \quad (4)$$

3. Results and Discussions

The classification model is trained using a dataset, and its performance is validated using K-fold cross-validation. Because the dataset has three classes with an unbalanced number, the Stratified K-fold Cross Validation method is used so that when the validation processes are performed, each fold has the same number of classes [20].

Table 6 shows the classification model's evaluation for each combination of hyperparameter tuning. This is done to find out which combination produces the best model performance. Based on Table 6, 12 combinations of hyperparameter tuning were produced, and the model performance for each combination can be seen in Table 7.

Table 7 Model Performance

Hyperparameters	Accuracy	Precision	Recall	F1-Score
Units: 32 Dropout: 0.4 Regularizer: L1	90.43%	90.20%	90.98%	90.70%
Units: 32 Dropout: 0.4 Regularizer: L2	94.35%	94.14%	94.43%	94.74%
Units: 32 Dropout: 0.5 Regularizer: L1	92.20%	92.12%	92.94%	92.64%
Units: 32 Dropout: 0.5 Regularizer: L2	95.24%	95.09%	95.15%	95.99%
Units: 64 Dropout: 0.4 Regularizer: L1	93.48%	93.87%	93.98%	93.09%
Units: 64 Dropout: 0.4 Regularizer: L2	94.66%	94.09%	94.65%	94.26%
Units: 64 Dropout: 0.5 Regularizer: L1	94.53%	94.41%	94.70%	94.24%
Units: 64 Dropout: 0.5 Regularizer: L2	93.83%	93.61%	93.55%	93.96%
Units: 96 Dropout: 0.4 Regularizer: L1	90.46%	90.91%	90.78%	90.35%
Units: 96 Dropout: 0.4 Regularizer: L2	91.58%	91.18%	91.41%	91.67%
Units: 96 Dropout: 0.5 Regularizer: L1	94.14%	94.05%	94.77%	94.97%
Units: 96 Dropout: 0.5 Regularizer: L2	95.81%	95.06%	95.67%	95.81%

As shown in Table 7, the ideal hyperparameter combination achieves accuracy, precision, recall, and f1-score of 95.24%, 95.09%, 95.15%, and 95.99%,

respectively. Units: 32, Dropout: 0.5, and Regularizer: L2 across all main assessment criteria; this mix produces the best results, especially in F1-Score, which acts as a balanced indicator between Precision and Recall. This is crucial for sentiment analysis tasks because erroneous classifications (false positives/false negatives) can exert a substantial influence [6][7].

A reduced Unit value, such as 32, enables the model to maintain sufficient representational ability to identify significant patterns while minimizing the number of parameters. On this dataset, larger Units (64, 96) tend to add unnecessary complexity, which can lead to overfitting or performance instability. Then, a dropout of 0.5 provides an ideal regularization level, helping the model prevent overfitting by turning off 50% of units during training. This results more consistently than a dropout of 0.4, which is too small and increases the model's noise sensitivity [7]. At last, the L2 regularizer enhances generalization by imposing a penalty on model big weights. Based on this data, the L2 regularizer performs better than the L1 since it maintains the weights small, but the distribution is more balanced, which is crucial for deep learning models, including hybrid CNN and Bi-LSTM [5][9].

In more detail, the training and validation accuracy graphs of the best models produced can be seen in Figure 4. Based on the graph, the training accuracy increases consistently, although it begins to show a slowdown in the increase after around the 60th epoch. Concurrently, the validation accuracy rises similarly to the training accuracy but at a marginally lower value. The validation accuracy aligns with the training accuracy trend, suggesting minimal overfitting in the model. Following the 60th epoch, the training and validation accuracy enhancement commences to stabilize. This indicates that the model has attained the optimal training phase and exhibits negligible enhancement in subsequent epochs. Hence, the training process concludes at the 83rd epoch.

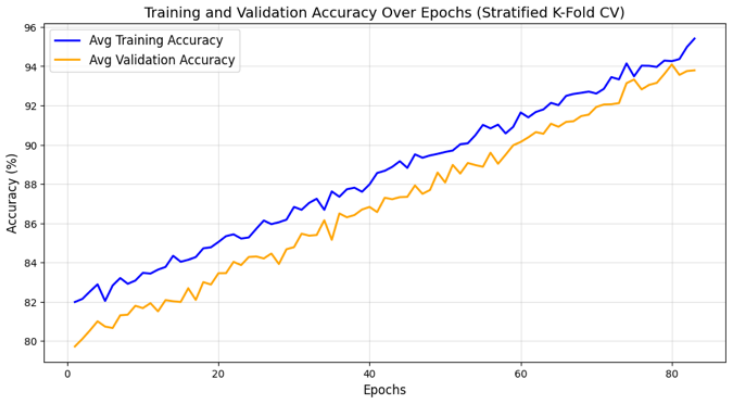

Figure 4. Graph of Training and Validation Accuracy

Additionally, the training and validation loss graphs of the best models produced can be seen in Figure 5. From the start to the end of the training, the training loss exhibits a constant decline based on the graph; early epochs show a notable fall and reach stability following epoch 60. Although it tended to be more erratic than the training loss, the validation loss also dropped significantly in the early epochs. After epoch 60, the validation loss also began to approach stability. Fluctuations in the validation loss at some points, especially before epoch 60, were caused by stratified k-fold cross-validation, which kept the class distribution balanced in each fold. The use of stratified k-fold helps maintain the consistency of model performance. This is reflected in the not-too-sharp decrease in a loss in the validation loss, even though the dataset is unbalanced.

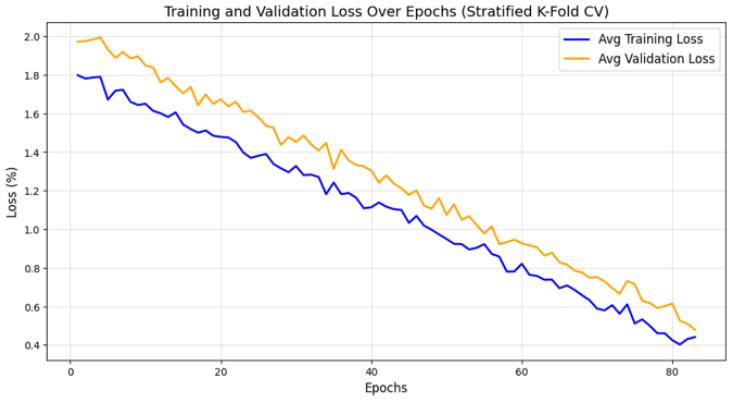

Figure 5. Graph of Training and Validation Loss

4. Conclusions

This study effectively illustrates the usefulness of a hybrid CNN and Bi-LSTM model in conducting sentiment analysis on Indonesian text datasets with three sentiment classes: positive, negative, and neutral. The findings indicate that the combination of CNN for feature extraction and Bi-LSTM for sequential information processing yields improved performance, with the ideal hyperparameters being Units: 32, Dropout: 0.5, and Regularizer: L2. Respectively, these hyperparameters generate performance with accuracy, precision, recall, and f1-score of 95.24%, 95.09%, 95.15%, and 95.99%. This approach effectively addresses the problems of extracting contextual and sequential traits in Indonesian text sentiment classification, which fewer solid techniques have always been limited. This research has likely been used in fields including customer feedback analysis, social media monitoring, and market sentiment evaluation, where accurate and quick sentiment analysis tools are vital. Adopting hybrid models highlights the requirement of using complementary architectures to overcome individual model constraints, optimizing performance for natural language processing applications. Future studies should investigate how this hybrid technique scales using more extensive, more varied datasets or how it adjusts to multilingual settings. Adding more pre-trained embeddings or complex regularizing approaches could enhance model resilience and generalization. Moreover, evaluating computation efficiency and real-time deployment options could offer interesting studies for general industrial applications.

References

[1] J. Deng and Y. Lin, "The Benefits and Challenges of ChatGPT: An Overview," *Frontiers in Computing and Intelligent Systems*, vol. 2, no. 2, pp. 81-83, 2022. https://doi.org/10.54097/fcis.v2i2.4465

[2] S. Rice, S. R. Crouse, S. R. Winter and C. Rice, "The advantages and limitations of using ChatGPT to enhance technological research," *Technology in Society*, vol. 76, 2024. https://doi.org/10.1016/j.techsoc.2023.102426

[3] A. Zein, "Dampak Penggunaan ChatGPT pada Dunia Pendidikan," Jurnal Informatika Utama, vol. 1, no. 2, pp. 19-24, 2023. https://doi.org/10.55903/jitu.v1i2.151

[4] M. Husnaini and L. M. Madhani, "Perspektif Mahasiswa terhadap ChatGPT dalam Menyelesaikan Tugas Kuliah," *Journal of Education Research*, vol. 5, no. 3, pp. 2655-2664, 2024. https://doi.org/10.37985/jer.v5i3.1047

[5] V. R. Kota and S. D. Munisamy, "High accuracy offering attention mechanisms based deep learning approach using CNN/bi-LSTM for sentiment analysis," *International Journal of Intelligent Computing and Cybernetics*, vol. 15, no. 1, 2022. https://doi.org/10.1108/IJICC-06-2021-0109

[6] Y. Zhou, Q. Zhang, D. Wang and X. Gu, "Text Sentiment Analysis Based on a New Hybrid Network Model," *Computational Intelligence Neuroscience* , vol. 6774320, 2022. https://doi.org/10.1155/2022/6774320

[7] M. Abbes, Z. Kechaou and A. M. Alimi, "A Novel Hybrid Model Based on CNN and Bi-LSTM for Arabic Multi-domain Sentiment Analysis," in *The 17th International Conference on Complex, Intelligent and Software Intensive Systems (CISIS-2023)*, Toronto, 2023. https://doi.org/10.1007/978-3-031-35734-3_10

[8] M. Umer, I. Ashraf, A. Mehmood, S. Kumari, S. Ullah and G. S. Choi, "Sentiment analysis of tweets using a unified convolutional neural network-long short-term memory network model," *Computational Intelligence*, vol. 37, no. 1, pp. 409-434, 2021. https://doi.org/10.1111/coin.12415

[9] P. K. Jain, V. Saravanan and R. Pamula, "A Hybrid CNN-LSTM: A Deep Learning Approach for Consumer Sentiment Analysis Using Qualitative User-Generated Contents," *ACM Transactions on Asian and Low-Resource Language Information Processing*, vol. 20, no. 5, pp. 1-15, 2021. https://doi.org/10.1145/3457206

[10] D. Atmajaya, A. Febrianti and H. Darwis, "Metode SVM dan Naive Bayes untuk Analisis Sentimen ChatGPT di Twitter," *The Indonesian Journal of Computer Science*, vol. 12, no. 4, pp. 2173-2181, 2023. https://doi.org/10.33022/ijcs.v12i4.3341

[11] Y. Akbar and T. Sugiharto, "Analisis Sentimen Pengguna Twitter di Indonesia Terhadap ChatGPT Menggunakan Algoritma C4.5 dan Naïve Bayes," *Jurnal Sains dan Teknologi*, vol. 5, no. 1, pp. 115-122, 2023. https://doi.org/10.55338/saintek.v4i3.1368

[12] V. R. Prasetyo and A. H. Samudra, "Hate speech content detection system on Twitter using K-nearest neighbor method," in *International Conference on Informatics, Technology, and Engineering 2021 (InCITE 2021)*, Surabaya, 2021. https://doi.org/10.1063/5.0080185

[13] R. CNBC Indonesia, "Raja Aplikasi Terbaru di RI, Ternyata Bukan WhatsApp-Instagram," *CNBC Indonesia*, 26 February 2024. [Online]. Available: https://www.cnbcindonesia.com/tech/20240226153650-37-517653/raja-aplikasi-terbaru-di-ri-ternyata-bukan-whatsapp-instagram. [Accessed 10 January 2025].

[14] M. Leotta, B. García and F. Ricca, "A family of experiments to quantify the benefits of adopting WebDriverManager and Selenium-Jupiter," *Information and Software Technology*, vol. 178, 2025. https://doi.org/10.1016/j.infsof.2024.107595

[15] M. A. Palomino and F. Aider, "Evaluating the Effectiveness of Text Pre-Processing in Sentiment Analysis," *Applied Sciences*, vol. 12, no. 17, 2022. https://doi.org/10.3390/app12178765

[16] M. F. Naufal and S. F. Kusuma, "Analisis Sentimen pada Media Sosial Twitter Terhadap Kebijakan Pemberlakuan Pembatasan Kegiatan Masyarakat Berbasis Deep Learning," *Jurnal Edukasi dan Penelitian Informatika*, vol. 8, no. 1, pp. 44-49, 2022. https://doi.org/10.26418/jp.v8i1.49951

[17] V. R. Prasetyo, G. Erlangga and D. A. Prima, "Analisis Sentimen untuk Identifikasi Bantuan Korban Bencana Alam berdasarkan Data di Twitter Menggunakan Metode K-Means dan Naive Bayes," *JurnalTeknologi Informasi dan Ilmu Komputer (JTIIK)*, vol. 10, no. 5, pp. 1055-1062, 2023. https://doi.org/10.25126/jtiik.20231057077

[18] H. D. Abubakar, M. Umar and M. A. Bakale, "Sentiment Classification: Review of Text Vectorization Methods: Bag of Words, Tf-Idf, Word2vec and Doc2vec," *Sule Lamido University Journal of Science & Technology*, vol. 4, no. 1, pp. 27-33, 2022. https://doi.org/10.56471/slujst.v4i.266

[19] O. Alharbi, "A Deep Learning Approach Combining CNN and Bi-LSTM with SVM Classifier for Arabic Sentiment Analysis," *(IJACSA) International Journal of Advanced Computer Science and Applications*, vol. 12, no. 6, pp. 165-172, 2021. https://dx.doi.org/10.14569/IJACSA.2021.0120618

[20] S. Ünalan, O. Günay, I. Akkurt, K. Gunoglu and H. O. Tekin, "A comparative study on breast cancer classification with stratified shuffle split and K-fold cross validation via ensembled machine learning," *Journal of Radiation Research and Applied Sciences*, vol. 17, no. 4, 2024. https://doi.org/10.1016/j.jrras.2024.101080

