Matrik: Jurnal Manajemen, Teknik Informatika, dan Rekayasa Komputer
Vol. 24, No. 2, March 2025, pp. 297∼308
ISSN: 2476-9843, accredited by Kemenristekdikti, Decree No: 200/M/KPT/2020
DOI: 10.30812/matrik.v24i1.4742

Mitigating Overfitting in Sentiment Analysis Insights from CNN-LSTM Hybrid Models

Susandri Susandri[1], Ahmad Zamsuri[1], Nurliana Nasution[1], Yoyon Efendi[2], Hiba Basim Alwan[3]
[1]Universitas Lancang Kuning, Pekanbaru, Indonesia
[2]University Utara Malaysia, Kedah, Malaysia
[3]University of Technology, Baghdad, Iraq

Article Info

Article history:

Received December 20, 2024
Revised February 26, 2025
Accepted March 06, 2025

Keywords:

Convolutional Neural Networks;
Hybrid Models;
Long Short-Term Memory;
Mitigating Overfitting;
Sentiment Analysis.

ABSTRACT

This study aims to improve sentiment analysis accuracy and address overfitting challenges in deep learning models by developing a hybrid model based on Convolutional Neural Networks and Long Short-Term Memory Networks. The research methodology involved multiple stages, starting with preprocessing a dataset of 5,456 rows. This process included removing duplicate data, empty entries, and neutral sentiments, resulting in 2,685 usable rows. Data augmentation expanded the training dataset from 2,148 to 10,740 samples to overcome data quantity limitations. Data transformation was carried out using tokenization, padding, and embedding techniques, leveraging Word2Vec and GloVe to produce numerical representations of textual data. The hybrid model demonstrated strong performance, achieving a training accuracy of 99.51%, validation accuracy of 99.25%, and testing accuracy of 87.34%, with a loss value of 0.56. Evaluation metrics showed precision, recall, and F1-Score values of 86%, 87%, and 86%, respectively. The hybrid model outperformed individual models, including Convolutional Neural Networks (70% accuracy) and Long Short-Term Memory Networks (81% accuracy). It also surpassed other hybrid models, such as the multiscale Convolutional Neural Network-Long Short-Term Memory Network, which achieved a maximum accuracy of 89.25%. The implications of this study demonstrate that the hybrid model based on Convolutional Neural Networks and Long Short-Term Memory Networks effectively improves sentiment analysis accuracy while reducing the risk of overfitting, particularly in small or imbalanced datasets. Future research is recommended to enhance data quality, adopt more advanced embedding techniques, and optimize model configurations to achieve better performance.

Corresponding Author:

Susandri Susandri, +628127617759,
Postgraduate of Computer Science,
Universitas Lancang Kuning, Pekanbaru, Indonesia,
Email: susandri@unilak.ac.id.

How to Cite:
S. Susandri, A. Zamsuri, N. Nasution, Y. Efendi, and H. Alwan, "Mitigating Overfitting in Sentiment Analysis Insights from CNN-LSTM Hybrid Models", *MATRIK: Jurnal Manajemen, Teknik Informatika, dan Rekayasa Komputer*, Vol.24, No.2, pp. 297-308, March, 2025.

1. INTRODUCTION

Sentiment analysis represents a pivotal domain within natural language processing (NLP), with its applications extending across diverse fields, including customer feedback evaluation, social media sentiment tracking, and strategic business decision-making [1]. However, deep learning models, such as convolutional neural networks (CNNs) and Long Short-Term Memory (LSTM) networks, frequently encounter the issue of overfitting, particularly when dealing with limited or imbalanced datasets. Overfitting reduces a model's generalization capability, hindering its adoption in practical environments [2–4].

A key challenge in sentiment analysis is ensuring the model can effectively generalize to previously unseen data [2]. Although CNNs and LSTMs possess exceptional capabilities in capturing complex patterns in data [5], they often result in overfitted models owing to their large parameter capacity [6, 7]. To address this challenge, a hybrid CNN-LSTM approach was introduced, leveraging the complementary strengths of CNNs for spatial feature extraction and LSTM for temporal sequence analysis.

Various approaches have been proposed to mitigate overfitting in sentiment analysis, including regularization, data augmentation, and transfer learning [8]. Recent studies have shown that the combination of CNN-LSTM outperforms single models, leveraging the complementary capabilities of both architectures [3, 4]. Although both [3] and [4] utilize hybrid CNN-LSTM models; they differ significantly in their methodologies and focus areas. [3] proposes a hybrid CNN-LSTM approach focusing on multilingual and multimodal sentiment analysis, employing datasets including text, images, and audio, and incorporating transfer learning techniques to enhance model generalization on small datasets. In contrast, [4] introduced a hybrid IChOA-CNN-LSTM model that integrates the Improved Chicken Swarm Optimization Algorithm (IChOA) with CNN-LSTM, focusing on social-media emotion recognition and utilizing datasets rich in emotional and linguistic diversity. These differences highlight the versatility of hybrid CNN-LSTM models in addressing various challenges in sentiment analysis. However, further research is required to optimize this combination, particularly in the context of overfitting mitigation.

The significance of text sentiment analysis has grown exponentially with the proliferation of social media platforms [9, 7]. Historically, lexicon-based techniques and traditional machine learning algorithms, including Naive Bayes and Support Vector Machines (SVM), have been the predominant methods employed in this domain [10, 11]. However, deep-learning-based approaches have gained widespread attention due to their higher accuracy with the increasing complexity of textual data. CNN and LSTM are two architectures that are widely adopted in text classification and sentiment analysis [12, 2]. Recently, hybrid approaches combining these two architectures have been applied to handle more complex sentiment analysis tasks [13, 1].

Several prior studies have demonstrated the success of hybrid CNN-LSTM approaches across various datasets. For instance, [14] achieved an accuracy of 87.2%, whereas [15] achieved 89.25% accuracy using MSCNN-LSTM. However, there remains a gap in the literature regarding overfitting mitigation in hybrid models, as most studies focus on the overall performance without addressing the direct impact of mitigation strategies on model generalization. Unlike previous studies that primarily focused on improving the overall accuracy of hybrid CNN-LSTM models, this research emphasizes developing and evaluating strategies specifically designed to mitigate overfitting and enhance generalization. While prior studies, such as [14] and [15], have demonstrated the effectiveness of hybrid models in achieving high accuracy, they have not systematically investigated the impact of overfitting mitigation techniques, such as dropout, batch normalization, and data augmentation, on model performance. This study addresses this gap by proposing a novel hybrid CNN-LSTM model that optimizes generalization and integrates advanced preprocessing, feature extraction, and layer configuration techniques.

The novelty of this research lies in its comprehensive approach to overfitting mitigation, which combines multiple strategies, such as dropout layers, batch normalization, and data augmentation, within a single hybrid model. In addition, this study introduces preprocessing innovations, including removing punctuation, duplicates, neutral sentiments, and short text, as well as using Word2Vec and GloVe embeddings for feature extraction. These contributions distinguish this study from previous studies and provide a more robust framework for sentiment analysis.

This study aims to develop a hybrid CNN-LSTM model with innovations in preprocessing, feature extraction, and model layer configuration. Preprocessing includes cleaning data from punctuation, duplicates, neutral sentiments, and short text. Feature extraction employs data augmentation techniques and embeddings based on Word2vec and GloVe. The model layer configuration included embedding, dropout, CNN, max-pooling, LSTM, and dense layers optimized for improved generalization.

The central aim of this research is to identify and assess effective strategies for mitigating overfitting in the hybrid CNN-LSTM model while simultaneously improving its ability to generalize to new data. The combination of strategies, such as dropout, batch normalization, and data augmentation, is expected to yield a sentiment analysis model with better generalization compared to individual models or hybrid models without these strategies. Applying overfitting mitigation strategies in the hybrid CNN-LSTM model will improve sentiment analysis performance, particularly in addressing overfitting and enhancing generalization on unseen data. The findings of this study are expected to have significant implications for the field of sentiment analysis, particularly in practical applications where generalization to unseen data is critical. By demonstrating the effectiveness of overfitting mitigation strategies in hybrid CNN-LSTM models, this study provides a foundation for developing more reliable and adaptable sentiment analysis systems.

These systems can be applied in real-world scenarios such as social media monitoring, customer feedback analysis, and business decision-making, where accurate and generalizable models are essential.

2. RESEARCH METHOD

This research seeks to mitigate overfitting in text-based sentiment analysis by creating a hybrid model that integrates architectures of convolutional neural networks (CNN) and Long Short-Term Memory (LSTM) synergistically. The model is engineered to harness the distinct advantages of each architecture: the CNN's proficiency in extracting spatial features, including local text patterns, and the LSTM's strength in modeling intricate temporal sequences. This research methodology includes dataset selection, data preprocessing, sentiment labeling, feature extraction, development of the hybrid CNN-LSTM model, and evaluation and analysis of results.

The dataset used in this study was sourced from the WhatsApp group "Forum DTC Riau," an online community consisting of 134 Daihatsu Taruna car owners in the Riau region. This dataset was chosen because of its unique characteristics, which include informal language, emoji usage, and local context, which are relevant for sentiment analysis. The conversation data were collected from March 16, 2023, to July 15, 2023, using an OPPO A15 device with Android 10 as the operating system.

The collected data consisted of unstructured text, including text messages, emojis, URLs, timestamp metadata, and encrypted message information. Pre-processing was performed to cleanse and standardize the data before analysis [1, 16]. This process included data extraction, where text data were separated from metadata such as timestamps, senders, and URLs using regular expressions, and translating text messages from Indonesian to English using the Google Translator API to facilitate sentiment analysis.

Irrelevant elements, such as punctuation, special characters, and emojis, were removed to improve data quality, and words in the text were lemmatized to their base forms to ensure consistency and enhance the model's generalization capability. Case folding was not performed because the Word2Vec and GloVe embeddings used in this study automatically handled case sensitivity, treating uppercase and lowercase words as the same entity. Additionally, stopword removal was intentionally omitted because of the informal nature of the WhatsApp dataset, which contains abbreviations, slang, and emojis that are crucial for sentiment analysis in conversational contexts. Removing stopwords can result in losing important contextual information, thereby reducing the model's accuracy. Messages with neutral sentiments were removed to focus on analyzing positive and negative sentiments. Sentiment labeling was performed to classify the data into three categories: positive, neutral, and negative [17, 18]. The translated text messages were analyzed using the nltk.sentiment.vader module to generate sentiment scores based on the compound score. The data were subsequently categorized according to the compound score thresholds: positive (compound score $\geq$ 0.05), negative (compound score $\leq$ -0.05), and neutral (compound score ranging between -0.05 and 0.05).

The labeled dataset was saved in CSV format with the filename "DTCRiau_sentimen.csv" for use in the subsequent stages. Feature extraction involves converting textual data into a numerical format, enabling it to be effectively processed by a machine learning model [19, 11]. This process included tokenization, where text was split into individual tokens using the Keras tokenization. Padding was employed to standardize the token length to align with the model's input requirements, with a maximum text length of 300 tokens. Text was represented as a numerical vector using word embedding methods, specifically Word2Vec or GloVe, to capture the semantic meanings of words.

To address the issue of imbalanced data, where the number of negative sentiment samples significantly outweighed the positive ones, we employed data augmentation techniques rather than oversampling methods, such as SMOTE or ADASYN. Data augmentation was selected because it is more suitable for textual data, allowing us to synthetically increase the number of samples without altering the original data distribution. Unlike SMOTE and ADASYN, which are primarily designed for numerical data and require additional transformations for text, data augmentation preserves the contextual integrity of text, making it more effective for sentiment analysis tasks. By generating synthetic data, we were able to expand the training dataset from 2,148 to 10,740 samples, thereby improving the ability of the model to generalize without introducing artifacts that could arise from oversampling techniques.

2.1. Proposed Model

The hybrid CNN-LSTM model was designed to address overfitting in text-based sentiment analysis [20, 21]. The model architecture, as illustrated in Figure 1 and Table 1, consists of several key layers. The model starts with an embedding layer that converts text into numerical vectors with an output dimension of (None, 100, 200), where 100 is the maximum token length, and 200 is the embedding vector dimension. This layer has a total of 1,672,000 parameters.

Next, a Dropout layer with a rate of 0.5 was incorporated to mitigate overfitting during training. The Conv1D layer employs 64 filters with a kernel size of three to extract spatial features from the text, yielding an output shape of (None, 98, 64) and 38,464 parameters. Following this, a Max Pooling layer with a pool size of two was applied to downscale the spatial dimensions, resulting in an output shape of (None, 49, 64).

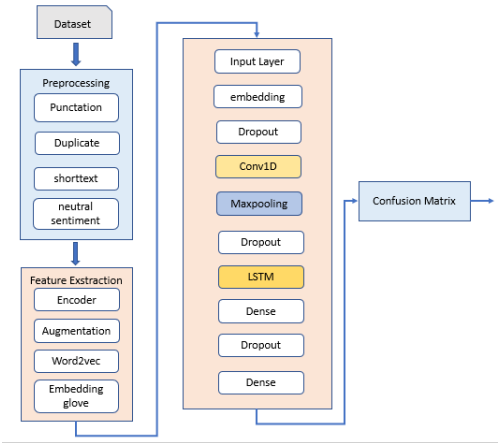

Figure 1. The architecture of the hybrid model is proposed in this study

A second dropout layer with a rate of 0.5 was introduced to enhance overfitting mitigation further. The LSTM layer, equipped with 128 memory units, was employed to analyze temporal patterns within the text sequence, generating an output shape of (None, 128) and a total of 98,816 parameters. The initial dense layer, featuring 128 units and a ReLU activation function, processed the extracted features, resulting in an output shape of (none, 128) and 16,512 parameters. To further combat overfitting, a third dropout layer at a rate of 0.5 was integrated.

The concluding layer is a dense layer comprising two units and a softmax activation function, tasked with final classification into two sentiment categories: positive and negative. This layer delivered an output shape of (None, 2) and encompassed 258 parameters. The model was optimized using the Adam algorithm over 20 epochs. The hyperparameters, including a padding size of 100 tokens, 128 LSTM units, and a dropout rate of 0.5, were meticulously tuned to strike a balance between model performance and overfitting prevention.

Table 1. Hyperparameters of the Proposed Hybrid Model

Model: ”sequential”		
Layer (type)	**Output Shape**	**Param #**
Embedding	(None, 100. 200)	1.672.000
Dropout	(None, 100. 200)	0
Conv1D	(None, 98. 64)	38464
Max_pooling	(None, 49. 64)	0
Dropout	(None, 49.64)	0
LSTM	(None, 128)	98816
Dense	(None, 128)	16512
Dropout	(None, 128)	0
Dense	(None, 2)	258

The classification outcomes were summarized using a confusion matrix, which served as the foundation for evaluating the model’s performance. In addition to accuracy, the evaluation metrics included precision, recall, and F1-Score. These metrics were benchmarked against prior studies that employed comparable methodologies to determine the effectiveness of the proposed hybrid Convolutional Neural Networks and Long Short-Term Memory Networks model. The following performance indicators were employed in this study to assess the model’s efficacy: Accuracy is defined as the ratio of correctly classified samples to the total number of samples, as shown in Equation 1:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{1}$$

Where TP represents True Positives, TN represents True Negatives, FP represents False Positives, and FN represents False Negatives. This metric provides a general overview of a model's performance by measuring the proportion of correct predictions. Accuracy is particularly useful for evaluating models on balanced datasets where the distribution of classes is relatively even. However, it may not be as effective for imbalanced datasets because it does not account for the distribution of errors across classes. Precision is the proportion of accurately identified positive samples out of all samples predicted as positive, as shown in Equation 2. This was calculated using the following formula:

$$Precision = \frac{TP}{TP + FP} \tag{2}$$

This metric focuses on the model's ability to avoid false positives, ensuring that the predicted positive samples are correct. High precision indicates a low rate of false positives, which is crucial in applications where false alarms are expensive. For example, in spam detection, high precision ensures that legitimate emails are not incorrectly classified as spam. However, precision alone does not account for false negatives, which may lead to incomplete evaluation of the model's performance. Recall is the proportion of accurately identified positive samples out of all actual positive samples, as shown in Equation 3. It is calculated using the formula:

$$Recall = \frac{TP}{TP + FN} \tag{3}$$

This metric measures the model's ability to identify all relevant positive samples, minimizing the number of false negatives. High recall is essential in scenarios where missing positive cases are undesirable, such as medical diagnoses or fraud detection. However, high recall may come at the cost of increased false positives, which can be problematic in certain applications. Therefore, recall should be evaluated with precision to provide a balanced assessment of the model's performance. The F1-Score is a comprehensive measure of the model's accuracy and is calculated as the harmonic mean of the precision and recall, as shown in Equation 4. It is calculated using the formula:

$$F1 - Score = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{4}$$

This metric balances the precision and recall, providing a single score that reflects the model's overall performance. This is particularly useful when dealing with imbalanced datasets, where one class significantly outweighs the other. The F1-Score ensures that both false positives and false negatives are considered, offering a more holistic evaluation of the model. By combining precision and recall, the F1-Score provides a robust metric for assessing models in scenarios where both types of errors are critical.

The performance outcomes of the hybrid CNN-LSTM model were benchmarked against those of individual models, including neural networks, RNN, LSTM, and CNN, to evaluate the advantages of the hybrid approach. This study is expected to significantly contribute to improving sentiment analysis accuracy while reducing the risk of overfitting using the hybrid CNN-LSTM approach.

We also compared the proposed model architecture with that of previous research, as described in Table 2. Each study employed a model architecture specifically designed to align with the unique characteristics of the data and incorporated distinct preprocessing techniques. For example, [22] used a CNN-LSTM model with preprocessing involving Corpus Cleaning, Text Segmentation, and Stop Word Removal on the IMDB dataset. This model used Word2vec for embedding and achieved an accuracy of 87.6%. [15] developed an MSCNN-LSTM model with word embedding as the text representation, achieving accuracies ranging from 85.5% to 89.25%. [23] used a CNN-LSTM model with preprocessing involving Stemming, Lemmatization, Stopword Removal, and Tokenization. The embeddings used included Word2vec, GloVe, TF-IDF, and FastText, resulting in 74% and 84% accuracy. [24] proposed a Hybrid CNN-RNN model with Word2Vec as the e mbedding, achieving an accuracy of 82.7% on balanced data. Other studies, [16] used a CNN-BiLSTM model with Sentiment Tagging and Wordvector embeddings, achieving an accuracy of 85.5%.

Table 2. Performance Metrics of Existing Models

Model	Embedding	F1-Score (%)	Accuracy (%)
Hybrid CNN-RNN (Balanced Data)[24]	Word2Vec	88.3	82.7
MSCNN-LSTM[15]	Word embedding	85.5	89.25
CNN-LSTM[22]	Word2vec	88.0	87.6
CNN-LSTM[23]	Word2vec, Glove, TF-IDF, FastText	89.0	82.1
CNN BiLSTM, CNN LSTM [16]	Sentiment Tagging, Wordvector	75.0	85.5

The developed Hybrid CNN-LSTM model consists of preprocessing, feature extraction, layer configuration, model testing, and evaluation stages. The innovative aspects of this study are primarily embodied in the feature extraction and layer configuration

phases. Preprocessing involved cleaning data from punctuation, duplicates, neutral sentiments, and short texts. The feature extraction stage included Splitting Encoder, Augmentation, Word2Vec, and the use of the 'glove.6B.200d.txt' embedding, which represents a novel approach compared to previous studies such as [15, 14, 23]. The layer configuration and model testing stage were designed with an embedding layer of size (100, 200), three dropout layers of sizes (100, 200), (49, 64), and (128), a CNN layer of size (98, 64), a max-pooling layer of size (49, 64), an LSTM layer of size (128), and two dense layers of size 128 and 2. The hybrid CNN-LSTM framework is shown in Figure 1. The layer configuration used represents a novel approach based on previous studies, such as [15, 22, 23] with their hybrid CNN-LSTM models.

3. RESULT AND ANALYSIS

3.1. Data Labeling

Data Labeling The preprocessed experimental data were labeled using the vaderSentiment library as a lexicon-based labeling approach [25]. Labeling resulted in three sentiment categories: positive, neutral, and negative. Before cleaning, the initial dataset consisted of 5,456 rows. The data cleaning results, including removing neutral sentiments and grouping of positive and negative sentiments, are presented in Table 3. This table shows the outcomes of data cleaning, which was conducted to prepare the dataset for the Hybrid CNN-LSTM model. The cleaning process involved the removal of duplicate data, empty data, and neutral sentiments. As a result, a cleaner dataset focusing on positive and negative sentiments was produced. Data Division The cleaned data were divided into training and testing sets with an 80:20 ratio (X_train, X_test, y_train, y_test) using random_state=0. The results of data division are presented in the form of graphs and distributions of text and word lengths, as shown in Figure 2.

Table 3. Data Cleaning Results for the Hybrid CNN LSTM Model

Cleaning	Data reading
dataset before removed duplicates	5456
dataset after removed duplicates	3493
dataset after drop empty :	3492
data set after removed netural sentiment	2685
sentiment positive	895
sentiment negative	1790

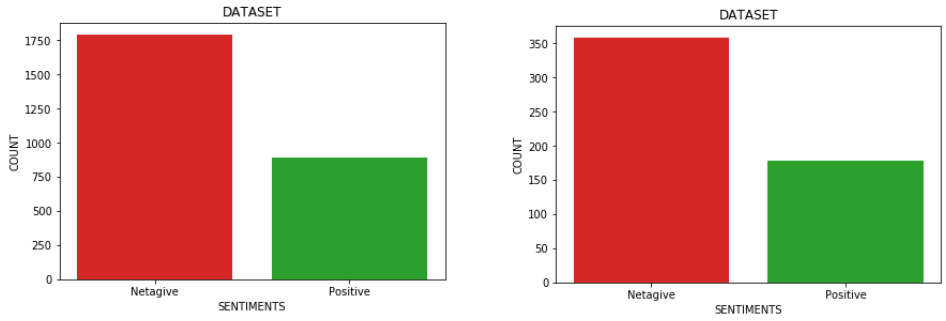

Figure 2. Graphs and distribution of test data and test data

3.2. Data Augmentation

Data Augmentation Augmentation has been performed to reduce overfitting by addressing the issue of a small training dataset [26]. Augmentation generated synthetic data from the original dataset without altering the original data [27]. The total training dataset size before augmentation was 2,148, which increased to 10,740 after the augmentation. The volume of positive sentiment data surged from 716 to 3,580, while negative sentiment data increased from 1,432 to 7,160. The outcomes of this augmentation process are detailed in Table 4.

Table 4. Test Data Augmentation Results

Augmentation train data	Results
Total training size before augmentation	(2148.2)
Total positive size before augmentation	(716.2)
Total negative size before augmentation	(1432.2)
Total training size after augmentation	-10740.2
Total positive size after augmentation	(3580.2)
Total negative size after augmentation	(7160.2)

Data Transformation Following the augmentation, data transformation was performed. Augmented data were combined with the test data, and positive sentiments were converted to 1, whereas negative sentiments were converted to 0. Subsequently, tokenization and padding were performed. The data transformation results are presented in Table 5. The testing process utilized Word2vec and glove.6B.200d as embeddings. The maximum sequence length was 100, with 5,388 unique words. The training data were encoded with a length of 10,740 and a width of 100, whereas the testing data were encoded with a length of 537 and a width of 100.

Table 5. Data Transformation Results

Data transformation	Results
Maximum sequence length	100
Total unique words	5388
Padded training data	(10740. 100)
Padded testing data	(537. 100)
Train label size	-10740
Test label size	(537. 2)

3.3. Model Training and Testing

The tokenized, padded, and Word2vec-embedded data were tested using the Hybrid CNN-LSTM model. The model architecture is shown in Figure 1. The training was conducted with the parameters n_splits=3, shuffle=True, random_state=0, batch_size=64, and epochs=35. The model had 1,231,650 trainable parameters. The training results showed that the model did not overfit, with minimal differences between training and validation accuracies. By epoch 31, the model attained a training accuracy of 99.51% and a validation accuracy of 99.25%. The training and validation results are shown in Figure 3.

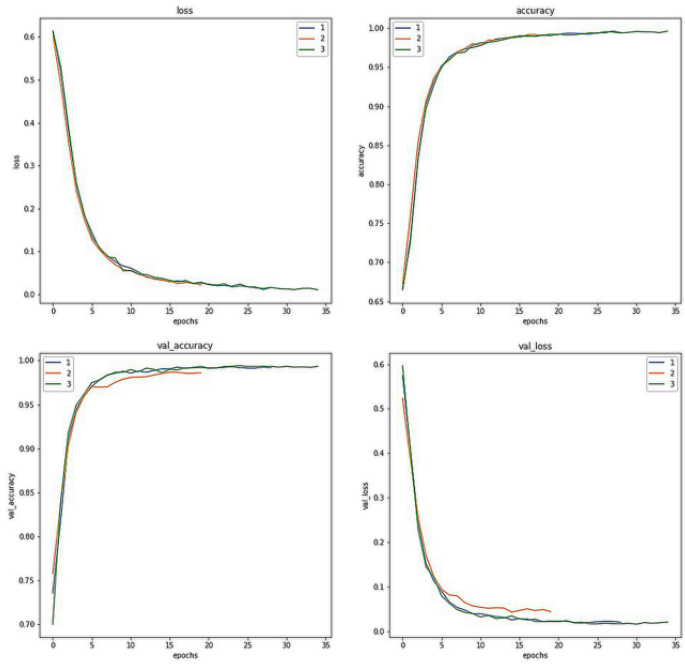

Figure 3. Hybrid CNN LSTM Training and Validation Results

Testing was conducted on 537 texts, resulting in an accuracy of 87.34% and a loss of 0.56. These results indicate that the model performed well in classifying sentiments based on the unseen test data. The test results are presented in Tables 6 and 7, respectively. Table 7 delineates the model's evaluation metrics, including the precision, recall, and F1-Score. For negative sentiments, the model achieved a precision of 90%, recall of 91%, and F1-Score of 90%, whereas, for positive sentiments, it recorded a precision of 83%, recall of 80%, and F1-Score of 81%. The comprehensive evaluation revealed an overall precision of 86%, a recall of 87%, and an F1-Score of 86%.

Table 6. Sentiment Analysis Model Test Results

Accuracy (%)	Loss
87.34	0.56

Table 7. Sentiment Analysis Model Evaluation Results

Label	Metrics		
	Precision	Recall	F1-Score
Negative	0.90	0.91	0.90
Positive	0.83	0.80	0.81
Average	0.86	0.87	0.86

Considering the testing outcomes, the Hybrid CNN-LSTM model demonstrated an accuracy of 87.34%, reflecting a substantial enhancement over single deep learning models, including LSTM (81%) and Bidirectional LSTM (81%). Furthermore, the model surpassed other deep learning architectures, such as neural networks (51%) and CNN (70%). The results are presented in Table 8. Comparison with Previous Research The results of this study were compared with those of previous studies, as shown in Table 2. Previous studies, such as [24], used a Hybrid CNN-RNN model with an accuracy of 82.7%,c[15] used an MSCNN-LSTM model with an accuracy of 89.25%, [22] used a CNN-LSTM model with an accuracy of 87.6%, [23] used a CNN-LSTM model with an accuracy of 82.1%, and [16] used a CNN-BiLSTM model with an accuracy of 85.5%.

This study attained an accuracy of 87.34%, signifying that the proposed Hybrid CNN-LSTM model is highly competitive with prior studies. Notably, the model proposed in [15], which utilized a Multi-Scale CNN-LSTM (MSCNN-LSTM) architecture, achieved a slightly higher accuracy of 89.25%. This improvement can be attributed to multiscale convolutions, which capture both local and global text patterns more effectively. However, our model focuses on mitigating overfitting and enhancing generalization, particularly for small and imbalanced datasets, which significantly contribute to the field.

Similarly, [14] achieved an accuracy of 87.6% using a CNN-LSTM model with preprocessing techniques, such as corpus cleaning, text segmentation, and stopword removal. Although their accuracy is comparable to ours, our model integrates advanced data augmentation and dropout layers specifically designed to address overfitting and improve generalization on unseen data. These strategies make our model more robust in practical applications where dataset sizes are often limited and imbalanced.

This study attained an accuracy of 87.34%, signifying that the proposed Hybrid CNN-LSTM model is highly competitive with prior studies. This study underscores the model's robust performance in sentiment analysis, with training outcomes exhibiting steady accuracy enhancements and a decline in loss. The training accuracy peaked at 99.51%, whereas the validation accuracy reached 99.25%.

Table 8. Detailed Results of Single Models and the Proposed Model

	Model	Accuracy (%)	Precision	Recall	F1-Score
Machine Learning	Random Forest	0.81	0.81	0.63	0.68
	Decision Tree	0.81	0.79	0.63	0.68
	Logistic Regression	0.76	0.70	0.53	0.57
	KNN	0.79	0.75	0.60	0.65
	Linear SVM	0.74	0.59	0.44	0.45
Deep Learning	Artificial Neural Network	0.74	0.46	0.40	0.39
	Neural Network	0.51	0.46	0.46	0.38
	Simple RNN	0.75	0.77	0.73	0.74
	LSTM	0.81	0.83	0.80	0.81
	CNN	0.70	0.68	0.68	0.68
Hybrid	**Proposed Model**	**0.87**	**0.86**	**0.87**	**0.88**

The testing phase yielded an accuracy of 87.34%, demonstrating the model's strong generalization capabilities for novel data. These findings are congruent with the research goals, which sought to elevate sentiment analysis accuracy using the hybrid CNN-LSTM methodology. The model successfully addressed the overfitting issue and demonstrated a good balance between the Precision and Recall. The outcomes met expectations, as the hybrid CNN-LSTM model surpassed the performance of individual deep learning models. However, some findings differed from expectations, such as slightly lower performance on positive sentiments than negative sentiments. This may be due to data variability and the labeling quality.

This study has several limitations, including potentially suboptimal data quality, particularly for truncated or improperly translated text messages. The use of glove.6B.200d embeddings, which may not include all unique words in the dataset, also poses a limitation. Additionally, the model configuration may not have been fully optimized, thus affecting performance. Future research should prioritize enhancing data quality by guaranteeing the accurate translation and thorough processing of all text messages. Using more comprehensive embeddings to handle a wider range of unique words can enhance the model performance. Optimizing the model configuration to improve the performance and generalization should also be considered.

4. CONCLUSION

The Hybrid CNN-LSTM model exhibited robust performance in the domain of sentiment analysis, achieving an accuracy of 87.34% on the test data. The training results showed an accuracy of 99.51% and a validation accuracy of 99.25%, exhibiting a favorable balance between Precision, Recall, and F1-Score; the model effectively mitigated the overfitting issue. The hybrid CNN-LSTM approach effectively improved sentiment analysis accuracy, particularly in mitigating overfitting. This model can be applied in various domains, such as product review analysis and social media monitoring. The limitations of this study include suboptimal data quality, particularly in truncated or improperly translated text messages, and the use of glove.6B.200d embeddings, which may not include all the unique words in the dataset. Additionally, the model configuration may not have been fully optimized, affecting performance. Future research should focus on improving the data quality, employing more comprehensive embeddings, and refining the model configuration. Investigating supplementary regularization techniques, including dropout or early stopping, may yield further benefits. This study offers significant insights into utilizing the Hybrid CNN-LSTM model in sentiment analysis tasks, particularly for mitigating overfitting. These findings underscore the potential for continued advancement in sentiment analysis.

5. ACKNOWLEDGEMENTS

We extend our heartfelt appreciation to all individuals and institutions who contributed to successfully completing this research. Furthermore, we would like to thank the WhatsApp group "DTC Riau Forum" participants for their contributions to the dataset used in this research. The participation and authenticity of the data are crucial to the research results. We also thank our colleagues for their insightful discussions, suggestions, and encouragement during this study.

6. DECLARATIONS

AUTHOR CONTIBUTION

Author Contributions Susandri Susandri initiated the research, designed the methodology, conducted experiments, analyzed data, and authored the manuscript. Ahmad Zamsuri contributed to data preprocessing and model development and provided critical feedback. Nurliana Nasution assisted in implementing the hybrid CNN-LSTM model and evaluating its outcomes. Yoyon Efendi and Hiba Basim Alwan provided valuable insights and feedback on the manuscript. All authors reviewed and approved the final version.

FUNDING STATEMENT

No external funding was received for this study. The authors acknowledge the support of Universitas Lancang Kuning, which provided the resources and facilities necessary for conducting it.

COMPETING INTEREST

The authors declare that they have no known competing financial interests or personal relationships that could have influenced the work reported in this study.

REFERENCES

[1] J. Khan, N. Ahmad, S. Khalid, F. Ali, and Y. Lee, "Sentiment and Context-Aware Hybrid DNN With Attention for Text Sentiment Classification," *IEEE Access*, vol. 11, no. 3, pp. 28 162–28 179, 2023, https://doi.org/10.1109/ACCESS.2023.3259107.

[2] N. A. Semary, W. Ahmed, K. Amin, P. Pławiak, and M. Hammad, "Improving sentiment classification using a RoBERTa-based hybrid model," *Frontiers in Human Neuroscience*, vol. 17, no. 12, pp. 1–10, 2023, https://doi.org/10.3389/fnhum.2023.1292010.

[3] M. S. Islam, M. N. Kabir, N. A. Ghani, K. Z. Zamli, N. S. A. Zulkifli, M. M. Rahman, and M. A. Moni, ""Challenges and future in deep learning for sentiment analysis: a comprehensive review and a proposed novel hybrid approach"," *Artificial Intelligence Review*, vol. 57, no. 3, pp. 1–79, 2024, https://doi.org/10.1007/s10462-023-10651-9.

[4] R. Geethanjali and A. Valarmathi, "A novel hybrid deep learning IChOA-CNN-LSTM model for modality-enriched and multilingual emotion recognition in social media," *Scientific reports*, vol. 14, no. 1, p. 22270, 2024, https://doi.org/10.1038/s41598-024-73452-2.

[5] A. Wahdan, S. Hantoobi, S. A. Salloum, and K. Shaalan, "A systematic review of text classification research based on deep learning models in Arabic language," *International Journal of Electrical and Computer Engineering*, vol. 10, no. 6, pp. 6629–6643, 2020, https://doi.org/10.11591/IJECE.V10I6.PP6629-6643.

[6] K. M. Hasib, S. Azam, A. Karim, A. A. Marouf, F. M. M. Shamrat, S. Montaha, K. C. Yeo, M. Jonkman, R. Alhajj, and J. G. Rokne, "MCNN-LSTM: Combining CNN and LSTM to Classify Multi-Class Text in Imbalanced News Data," *IEEE Access*, vol. 11, no. 9, pp. 93 048–93 063, 2023, https://doi.org/10.1109/ACCESS.2023.3309697.

[7] N. Zhang, J. Xiong, Z. Zhao, M. Feng, X. Wang, Y. Qiao, and C. Jiang, "Dose My Opinion Count? A CNN-LSTM Approach for Sentiment Analysis of Indian General Elections," *Journal of Theory and Practice of Engineering Science*, vol. 4, no. 05, pp. 40–50, 2024, https://doi.org/10.53469/jtpes.2024.04(05).06.

[8] A. Sungheetha, "TransCapsule Model for Sentiment Classification," *Journal of Artificial Intelligence and Capsule Networks*, vol. 02, no. 03, pp. 163–169, 2020, https://doi.org/10.36548/jaicn.2020.3.003.

[9] D. Chai, W. Wu, Q. Han, W. Fei, and J. Li, "Description based text classification with reinforcement learning," in *37th International Conference on Machine Learning, ICML 2020*, vol. 119, no. 1, 2020, pp. 1348–1359.

[10] P. Sudhir and V. Deshakulkarni, "Comparative study of various approaches , applications and classifiers for sentiment analysis," *Global Transitions Proceedings*, vol. 2, no. 2, pp. 205–211, 2021, https://doi.org/10.1016/j.gltp.2021.08.004.

[11] L. Irfan, S. Hussain, M. Ayoub, Y. Yu, and A. Khan, "A Comparative Analysis of Social Communication Applications using Aspect Based Sentiment Analysis," *Pakistan Journal of Engineering and Technology, PakJET*, vol. 5, no. 3, pp. 44–50, 2022, https://doi.org/10.51846/vol5iss3pp44-50.

[12] O. Iparraguirre-villanueva, A. Alvarez-risco, J. Luis, H. Salazar, S. Beltozar-clemente, J. Zapata-paulini, A. Y. Jaime, and M. Cabanillas-carbonell, "The Public Health Contribution of Sentiment Analysis of Monkeypox Tweets to Detect Polarities Using the CNN-LSTM Model number," *vaccines*, vol. 11, no. 312, pp. 1–12, 2023, https://doi.org/10.3390/vaccines11020312.

[13] A. Gupta and P. Agarwal, "Integrating CRM and ERP Insights for Optimized Product Development Using CNN-LSTM Hybrid Models," *International Journal of Computer Trends and Technology*, vol. 72, no. 8, pp. 91–97, 2024, https://doi.org/10.14445/22312803/IJCTT-V72I8P113.

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

[15] N. Jin, J. Wu, X. Ma, K. Yan, and Y. Mo, "Multi-task learning model based on Multi-scale CNN and LSTM for sentiment classification," *IEEE Access*, vol. 8, no. 4, pp. 77 060–77 072, 2020, https://doi.org/10.1109/ACCESS.2020.2989428.

[16] S. Soumya and K. V. Pramod, "Hybrid Deep Learning Approach for Sentiment Classification of Malayalam Tweets," *(IJACSA) International Journal of Advanced Computer Science and Applications*, vol. 13, no. 4, pp. 891–899, 2022, https://doi.org/10.14569/IJACSA.2022.01304103.

[17] A. Mohta, A. Jain, A. Saluja, and S. Dahiya, "Pre-processing and emoji classification of whatsapp chats for sentiment analysis," in *Proceedings of the 4th International Conference on IoT in Social, Mobile, Analytics and Cloud, ISMAC 2020*, 2020, pp. 514–519, https://doi.org/10.1109/I-SMAC49090.2020.9243443.

[18] S. Wu, K. Roberts, S. Datta, J. Du, Z. Ji, Y. Si, S. Soni, Q. Wang, Q. Wei, Y. Xiang, B. Zhao, and H. Xu, "Deep learning in clinical natural language processing: A methodical review," pp. 457–470, 2020, https://doi.org/10.1093/jamia/ocz200.

[19] L. Cabral, J. Monteiro, J. Franco da Silva, C. Mattos, and P. Mourão, "FakeWhastApp.BR: NLP and Machine Learning Techniques for Misinformation Detection in Brazilian Portuguese WhatsApp Messages," in *Proceedings of the 23rd International Conference on Enterprise Information Systems*, vol. 1, 2021, pp. 63–74, https://doi.org/10.5220/0010446800630074.

[20] 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," vol. 20, no. 5, pp. 1–15, 2021, https://doi.org/10.1145/3457206.

[21] J. Sun, R. Jin, X. Ma, J.-y. Park, K.-a. Sohn, and T.-s. Chung, "Gated Convolutional Neural Networks for Text Classification," in *Advances in Computer Science and Ubiquitous Computing*, J. J. Park, S. J. Fong, Y. Pan, and Y. Sung, Eds. Singapore: Springer Singapore, 2021, pp. 309–316, https://doi.org/10.1007/978-981-15-9343-7_43.

[22] Y. Zhou, Q. Zhang, D. Wang, and X. Gu, "Text Sentiment Analysis Based on a New Hybrid Network Model," *Computational Intelligence and Neuroscience*, vol. 2022, no. 12, pp. 1–15, 2022, https://doi.org/10.1155/2022/6774320.

[23] L. Khan, A. Amjad, K. M. Afaq, and H.-t. Chang, "Deep Sentiment Analysis Using CNN-LSTM Architecture of English and Roman Urdu Text Shared in Social Media," *Applied Sciences*, vol. 12, no. 6, pp. 1–18, 2022, https://doi.org/10.3390/app12052694.

[24] S. Riyadi, A. Divayu Andriyani, and S. Noraini Sulaiman, "Improving Hate Speech Detection Using Double-Layers Hybrid CNN-RNN Model on Imbalanced Dataset," *IEEE Access*, vol. 12, no. 10, pp. 159 660–159 668, 2024, https://doi.org/10.1109/ACCESS.2024.3487433.

[25] S. Susandri, S. Defit, and M. Tajuddin, "Enhancing Text Sentiment Classification with Hybrid CNN-BiLSTM Model on WhatsApp Group," *Journal of Advances in Information Technology*, vol. 15, no. 3, pp. 355–363, 2024, https://doi.org/10.12720/jait.15.3.355-363.

[26] G. Chao, J. Liu, M. Wang, and D. Chu, "Data augmentation for sentiment classification with semantic preservation and diversity," *Knowledge-Based Systems*, vol. 280, no. 11, p. 111038, 2023, https://doi.org/10.1016/j.knosys.2023.111038.

[27] J. Chen, Z. Yang, and D. Yang, "MixText: Linguistically-Informed Interpolation of Hidden Space for Semi-Supervised Text Classification," *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, vol. 1, no. 1, pp. 2147–2157, 2020, https://doi.org/10.18653/v1/2020.acl-main.194.

[This page intentionally left blank.]