VOL. 11. NO. 3 FEBRUARY 2026
P-ISSN: 2685-8223 | E-ISSN: 2527-4864
DOI: 10.33480 /jitk.v11i3.7488

JITK (JURNAL ILMU PENGETAHUAN DAN TEKNOLOGI KOMPUTER)

HYBRIDIZATION OF FASTTEXT-BLSTM AND BERT FOR ENHANCED SENTIMENT ANALYSIS ON SOCIAL MEDIA TEXTS

Jasmir Jasmir[1*]; Maria Rosario[2]; Irawan Irawan[3]; Agus Siswanto[3]; Tiko Nur Annisa[1]

Master of Information System Study Program[1]
Information System Study Program[2]
Informatics Engineering Study Program[3]
Universitas Dinamika Bangsa, Jambi, Indonesia[1,2,3]
https://unama.ac.id[1,2,3]
ijay_jasmir@yahoo.com*, diamar_ros@yahoo.com, irend.irawan.irend@gmail.com,
agussiswanto@unama.ac.id, tikonurannisa10@gmail.com

(*) Corresponding Author
(Responsible for the Quality of Paper Content)

Abstract—*The development of internet technology and social media has driven the increasing use of sentiment analysis to understand public opinion. This study aims to improve the classification performance of sentiment analysis by proposing a hybrid model that combines FastText-BLSTM and BERT. The dataset used consists of 900 Indonesian-language Netflix app user reviews obtained through crawling using Google Play Scraper. The research stages include text preprocessing, feature extraction using FastText and BERT, and classification using BLSTM, which are then combined in a concatenation layer to produce a richer feature representation. Experimental results show that the FastText-BLSTM-BERT hybrid model provides the best performance with an accuracy of 94.22%, a precision of 95.98%, a recall of 95.68%, and an F1-score of 95.83%. This achievement is superior to the single models of FastText-BLSTM and BERT. The main novelty of this research lies in the integration of contextual embeddings from BERT with subword-level semantic and sequential representations from FastText-BLSTM, which has not been extensively explored in prior studies on Indonesian sentiment analysis. This hybridization demonstrates significant improvement in model generalization and robustness for low-resource language texts.*

Keywords: *BERT, BLSTM, FastText, Hybrid Model, Sentiment Analysis.*

Intisari—*Perkembangan teknologi internet dan media sosial telah mendorong meningkatnya penggunaan analisis sentimen untuk memahami opini publik. Penelitian ini bertujuan untuk meningkatkan kinerja klasifikasi analisis sentimen dengan mengusulkan model hibrida yang menggabungkan FastText-BLSTM dan BERT. Dataset yang digunakan terdiri dari 900 ulasan pengguna aplikasi Netflix berbahasa Indonesia yang diperoleh melalui perayapan menggunakan Google Play Scraper. Tahapan penelitian meliputi prapemrosesan teks, ekstraksi fitur menggunakan FastText dan BERT, dan klasifikasi menggunakan BLSTM, yang kemudian digabungkan dalam lapisan konkatenasi untuk menghasilkan representasi fitur yang lebih kaya. Hasil eksperimen menunjukkan bahwa model hibrida FastText-BLSTM-BERT memberikan kinerja terbaik dengan akurasi 94,22%, presisi 95,98%, recall 95,68%, dan F1-score 95,83%. Pencapaian ini lebih unggul dibandingkan model tunggal FastText-BLSTM dan BERT. Kebaruan utama penelitian ini terletak pada integrasi embedding kontekstual dari BERT dengan representasi semantik dan sekuensial tingkat subkata dari FastText-BLSTM, yang belum banyak dieksplorasi dalam studi-studi sebelumnya tentang analisis sentimen bahasa Indonesia. Hibridisasi ini menunjukkan peningkatan yang signifikan dalam generalisasi dan ketahanan model untuk teks bahasa dengan sumber daya terbatas.*

Kata Kunci: *Analisis Sentimen, BERT, BLSTM, FastText, Model Hibrida.*

Accredited Rank 2 (Sinta 2) based on the Decree of the Dirjen Penguatan RisBang Kemenristekdikti No.225/E/KPT/2022, December 07, 2022. Published by LPPM Universitas Nusa Mandiri

VOL. 11. NO. 3 FEBRUARI 2026
P-ISSN: 2685-8223 | E-ISSN: 2527-4864
DOI: 10.33480/jitk.v11i3.7488

INTRODUCTION

Currently, technological developments are experiencing a very rapid surge, especially since the emergence of the internet. The advancement of technology has caused major changes in various sectors, including information and communication[1]. he significant increase in internet technology has expanded the reach of information distribution. One aspect that supports this increase is social media, where users not only function as recipients of information but also as creators of information. The rise in the number of internet users in Indonesia is driven by the conveniences offered by social media and the internet, which allow people to access and exchange information quickly. The utilization of data from social media has become an innovative approach that provides alternative data sources beyond traditional survey-based methods [2],[3],[4]. Data collection through social media offers high efficiency in terms of cost, time, and accessibility, while producing real-time and more nuanced data that reflect genuine public opinions[5]. his process of understanding and analyzing public attitudes expressed in online texts is commonly known as sentiment analysis [6][7][8].

Sentiment analysis, a subfield of Natural Language Processing (NLP), employs machine learning to identify emotional tone and classify text as positive, neutral, or negative [9] [10]. Since computers in Natural Language Processing (NLP) do not naturally understand text, various techniques are used to convert words into numerical vectors for easier machine interpretation. The ongoing research into word vector representation is crucial because it directly affects the accuracy and performance of learning models. This word representation technique falls under the scope of feature engineering, which is particularly challenging for textual data due to its unstructured nature. One of the most popular feature engineering strategies for textual data is word embedding. Previous studies have investigated different aspects of word embeddings to enhance text representation. Guo and Caliskan [11] revealed that contextualized embeddings, while powerful, can still inherit and amplify human-like biases, which may reduce fairness and interpretability in sentiment classification tasks. Zhuang et al. [12] proposed an out-of-vocabulary (OOV) embedding learning mechanism based on reading comprehension, yet their approach was designed mainly for English datasets and required substantial computational resources, making it less suitable for low-resource languages such as Indonesian. Meanwhile, Jasmir et al. [13] demonstrated that word embedding integration with Bidirectional Long Short-Term Memory (BLSTM) improved classification accuracy, but their study focused only on single embedding models without exploring deeper contextual representations like those produced by Transformer-based models.

This word embedding feature is collaborated with the classification method. There are many types of classifiers that are commonly used to classify sentiment analysis. The methods that are often used are machine learning[14], [15],[16],[17] and *deep learning*,[18][19]. In this study, the type of method used is the deep learning method, namely the BERT (Bidirectional Encoder Representations from Transformers) method [20] which is part of Transformer-Based Models. The characteristic of BERT is the ability to process the entire sequence of words in a sentence simultaneously [21]. This allows BERT to understand the context of a word based on the words that come before and after it, resulting in a richer and more accurate understanding of meaning. This BERT model uses self-attention to understand the relationship between elements in a sequence. The self-attention mechanism in Transformer allows BERT to theoretically connect distant words in a text more effectively than other models[22]. Each word can immediately "notice" other words, regardless of their distance. With this mechanism, the model can overcome long-range dependencies, understand context more accurately, and produce better predictions.

Another deep learning model used is BLSTM [23], BLSTM is a type of Recurrent Neural Network (RNN)[24]. BLSTM consists of two LSTM layers: one processes the sequence from front to back (forward), and the other processes the sequence from back to front (backward). The outputs of these two layers are then combined to obtain a contextual representation. Processing in RNN is sequential, meaning that information is processed step by step through the input sequence. BLSTM has the ability to access past and future information simultaneously, allowing BLSTM to capture richer and more accurate contextual dependencies than unidirectional LSTM. BLSTM often achieves higher accuracy in various sequential processing tasks and is effective in learning patterns and dependencies in long sequential data. The Word Embedding feature collaborated here is FastText[25][26]. FastText is a powerful and flexible tool in NLP, especially due to its ability to handle Out-of-Vocabulary (OOV) words and understand the morphological structure of words through the use of n-gram characters[27]. Its speed and efficiency also make it a popular choice for a variety of applications. Some previous studies that have discussed overlapping with this study are Nafaa Haffar and Mounir Zrigui[28] who used BERT and FastText to improve the BLSTM model on Arabic Text. Evaluation on the Arabic TimeBank corpus using FastText and BERT embeddings (rather than Skip-Gram) validated the proposed model's effectiveness.

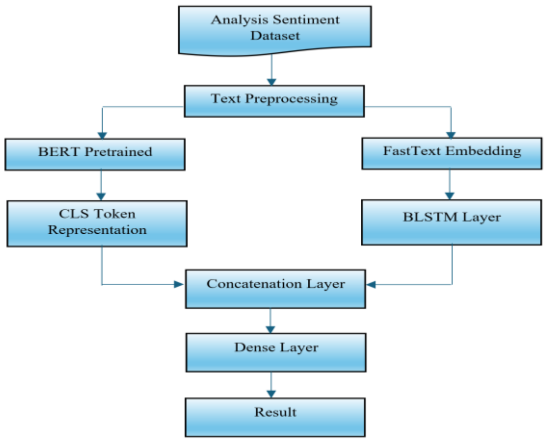
The model's power comes from its blend of BiLSTM-derived temporal features and spatial features. Another study by Nafaa Haffar[29] introduced a novel artificial neural network architecture combining BERT, POS features, event position, CNN, layered BiLSTM, and an attention mechanism to classify temporal relationships between events in Arabic sentences. Leveraging a combination of contextual representations and linguistic features, the model achieved an F1 score of 89% on the Ara-TimeBank corpus, surpassing previous research. Subsequently, a study by Hakan Gunduz [30] explored the performance comparison of two commonly used text representation models—Bidirectional Encoder Representations from Transformers (BERT) and FastText—combined with Long Short-Term Memory (LSTM) and Gradient Boosting Machines (GBM) classifiers. In this research, BERT and FastText were utilized for feature extraction, and their predictive performance was assessed using LSTM and GBM models. The experimental results consistently demonstrated that BERT representations significantly outperformed FastText, with the highest accuracy of 0.745 achieved by a fine-tuned BERT model combined with LSTM. Another study by Shreyashree [31]examined social media data related to COVID-19 quarantine using multi-class text classification across 15 categories. A hybrid LSTM-GRU model was developed using GloVe and BERT word embeddings. The results showed that the pre-trained BERT-hybrid performed slightly better than the GloVe-hybrid, but the tuned BERT model outperformed the tuned BERT model by 3%. With more epochs, the hybrid model is expected to outperform the tuned BERT model.

Then the next study is Nithya K et al[32] Then the next study is Nithya K et al [25] conducted a study of deep learning classifiers with BERT, Optimize CNN (OPCNN), FastText Information Gain and Ant Colony (FIAC), which were adjusted with BLSTM for rumor text classification. This study found that the feature vectors generated using the BERT-OPCNN and FIAC embedding models were classified using a customized Bi-LSTM. The experiments were conducted on both balanced and imbalanced datasets and compared with existing methods. The evaluation results indicate that the proposed FIAC embedding combined with BERT-OPCNN delivers superior performance compared to all previously established approaches when using the tailored Bi-LSTM classifier. However, most existing studies are focused on high-resource languages, while research applying such hybridizations to Indonesian text remains very limited.

The research gap lies in how to effectively combine FastText, BLSTM, and BERT within the context of Indonesian sentiment analysis, considering the language's unique morphology, affixation patterns, and limited availability of annotated datasets. The challenge is to integrate FastText's subword-level embeddings, BLSTM's sequential pattern learning, and BERT's contextual understanding to generate richer and more representative features for classification. Therefore, this study proposes a hybrid FastText-BLSTM-BERT model designed to enhance sentiment classification accuracy for Indonesian-language social media texts. The experiment uses user reviews of the Netflix application, chosen due to its widespread popularity and active user engagement, making it a relevant domain for sentiment evaluation. By bridging the gap between subword-level, sequential, and contextual representations, this study aims to demonstrate a more robust and generalizable approach for sentiment analysis in low-resource languages such as Indonesian.

MATERIALS AND METHODS

In order for this research to achieve optimal results, a series of systematic stages were designed to produce an accurate and reliable hybrid model for sentiment classification. The stages are compiled in the form of a research framework, as shown in Figure. 1.



Source : (Research Result, 2025)
Figure 1. Research Framework

VOL. 11. NO. 3 FEBRUARI 2026
P-ISSN: 2685-8223 | E-ISSN: 2527-4864
DOI: 10.33480/jitk.v11i3.7488

Figure 1 illustrates the workflow of the proposed hybrid model for sentiment analysis, combining the strengths of two major approaches in Natural Language Processing (NLP):

1. A Transformer-based model (BERT) for contextualized embedding extraction.
2. A Recurrent Neural Network-based model (BLSTM) with FastText embedding for sequential and subword-level representation.

This dual-path framework aims to leverage the contextual depth of BERT and the sequential learning strength of BLSTM to improve sentiment classification accuracy in Indonesian-language social media texts.

Dataset Description.

This is the starting point of the process. This dataset contains the text of Netflix application user comments. The dataset was obtained through a data collection process carried out by crawling. We utilize the Google Play Scraper Python library. To crawl data, the application ID from which the data will be taken is first required. In this case, Netflix has the ID 'com.netflix.mediaclient'. Furthermore, the selection of the language in the review is an important step, where this study only considers reviews in Indonesian. After selecting the language, the selection of reviews is based on the score. In this study, the reviews taken have a score range of 1 to 5. Furthermore, the order of the reviews used is Most Relevant. The amount of data to be taken also needs to be determined. The data obtained has several attributes, including: reviewId, username, userImage, content, score, thumbsUpCount, reviewCreatedVersion, at, replyContent, answeredAt, and appVersion. However, not all of these attributes are needed for this study. Therefore, irrelevant or unused attributes are removed to simplify the data. There are 4 attributes that will be used, namely Username, Score, Date and Content. The amount of data we can crawl is 900 records, with two classes, namely positive sentiment and negative sentiment. The amount of data from positive sentiment is 354 positive and the amount of data from negative sentiment is 546 negative. The dataset can be seen in Tabel 1.

Tabel 1. Dataset Snippet (Dataset In Indonesian Language)

Score	Date	Time	Content
4	07/09/2024	11.02	Kenapa film Wakanda Forever tidak bisa diputar
1	07/09/2024	10.33	Video tidak bisa diputar, tidak bisa menonton semua video. Padahal sudah berlangganan premium.
1	07/09/2024	08.29	Baru aja di Indo beli paket setahun, tapi pas mau nonton di luar negeri, seperti di Timur Tengah, tidak bisa.
1	07/09/2024	02.53	Kenapa tidak bisa putar film, cuma keluar layar item tiba-tiba langsung restart sampai perangkatnya ikut.
1	06/09/2024	17.39	Ini gimana sih aplikasinya, sudah tahu banyak yang kasih saran, bukan diperbaiki malah dibiarkan.
2	06/09/2024	14.10	Sekelas Disney saja filmnya tidak lengkap, masih ada beberapa film Disney yang tidak ada di sini.
1	06/09/2024	13.22	Percuma download dan bayar untuk sebulan tapi tidak bisa ditonton, malah muter terus. Rugi.
1	06/09/2024	10.42	Apa-apaan ini nonton tidak ada gambarnya tiba-tiba. Kebiasaan loh.
2	06/09/2024	07.44	Kenapa sih harus berlangganan, aku sudah belain download aplikasi ini karena di aplikasi lain tidak ada film yang aku cari.
1	06/09/2024	06.10	Gunain bikin apa sih berlangganan semua, tidak ada yang gratis. Kalau begitu bikin APK lah, mikir aneh.
2	06/09/2024	05.03	Kenapa sering lag ya, padahal sinyal dan kuota lancar, aplikasi sekelas Disney kok bermasalah.
5	06/09/2024	01.21	Saya sudah berlangganan tahunan, belum ada satu bulan kenapa black screen? Bagaimana mengatasinya?
5	05/09/2024	14.11	Punya saya berlangganan satu tahun tapi kok tidak bisa ditampilkan di televisi ya?
1	05/09/2024	13.32	Layarnya ketutupan subtitle, kecilin dong subtitle-nya, harusnya bisa custom ukuran subtitle.
2	05/09/2024	09.24	Filmnya ada yang dihapus padahal mau ditonton. Padahal dulu juga masih ada.
5	05/09/2024	08.19	Respon cepat terhadap kendala pelanggan, mantap.

Source : (Research Result, 2025

Text Preprocessing

Preprocessing aims to clean and standardize the textual data prior to model input. The main stages include:

1. Normalization: Correction of non-standard words and typographical errors using custom normalization mapping.
2. Tokenization: Splitting sentences into word tokens using the *Indonesian spaCy tokenizer*.
3. Lowercasing: Conversion of all tokens to lowercase for consistency.
4. Punctuation Removal: Elimination of punctuation marks and non-alphanumeric symbols.
5. Stopword Removal: Elimination of common Indonesian stopwords using the Sastrawi library.

6. Stemming/Lemmatization: Conversion of words into their root forms using the Sastrawi stemmer, a standardized Indonesian NLP tool.
7. Labeling: Sentiment labeling was performed manually using a lexicon-based approach adapted from the Indonesian sentiment lexicon. Two annotators independently labeled the dataset, and Cohen's Kappa (κ = 0.87) was used to ensure high inter-annotator agreement, indicating reliable labeling consistency.

Normalization

Before starting, it is a good idea to normalize for non-standard words or typos so that further processing is more accurate. The normalization process can be seen in table 2.

Table 2. Normalization Process

Before Normalization	After normalization
"Ini gimana si aplikasinya, sudah tau banyak yang ngasih saran, bukan diperbaiki, malah dibiarkan seperti itu. Malah kebelet minta langganan 1 tahun lagi, yang perbulan saja seperti ini. Gimana si managementnya, gak lihat apa, jangan untungnya aja yang dikelolah."	"Ini bagaimana sih aplikasinya, sudah tahu banyak yang memberi saran, bukan diperbaiki, malah dibiarkan seperti itu. Malah kebelet minta langganan 1 tahun lagi, yang perbulan saja seperti ini. Bagaimana sih managementnya, tidak lihat apa, jangan untungnya saja yang dikelola."

Source : (Research Result, 2025)

Tokenization

Breaking text into word units (tokens) so that they can be analyzed one by one by the NLP model. The Tokenization process can be seen in Table 3.

Table 3 . Tokenization Process

Normalization Sentence	Tokenization
"Ini bagaimana sih aplikasinya, sudah tahu banyak yang memberi saran, bukan diperbaiki, malah dibiarkan seperti itu. Malah kebelet minta langganan 1 tahun lagi, yang perbulan saja seperti ini. Bagaimana sih managementnya, tidak lihat apa, jangan untungnya saja yang dikelola."	"ini", "bagaimana", "sih", "aplikasinya", "sudah", "tahu", "banyak", "yang", "memberi", "saran", "bukan", "diperbaiki", "malah", "dibiarkan", "seperti", "itu", "malah", "kebelet", "minta", "langganan", "1", "tahun", "lagi", "yang", "perbulan", "saja", "seperti", "ini", "bagaimana", "sih", "managementnya", "tidak", "lihat", "apa", "jangan", "untungnya", "saja", "yang", "dikelola"

Source : (Research Result, 2025)

Lowercasing

Equalizing all letters to lowercase so that there is no difference in meaning between words such as "Ini" and "ini". This makes the analysis consistent. The lowercasing process can be seen in table 4.

Table 4. Lowercasing Process

Normalization Sentence	Lowercasing
"Ini bagaimana sih aplikasinya, sudah tahu banyak yang memberi saran, bukan diperbaiki, malah dibiarkan seperti itu. Malah kebelet minta langganan 1 tahun lagi, yang perbulan saja seperti ini. Bagaimana sih managementnya, tidak lihat apa, jangan untungnya saja yang dikelola."	"ini bagaimana sih aplikasinya, sudah tahu banyak yang memberi saran, bukan diperbaiki, malah dibiarkan seperti itu. malah kebelet minta langganan 1 tahun lagi, yang perbulan saja seperti ini. bagaimana sih managementnya, tidak lihat apa, jangan untungnya saja yang dikelola."

Source : (Research Result, 2025)

Punctuation Removal

Removing punctuation such as commas and periods that do not add significant meaning in most text analysis. Table 5 is the Punctuation Removal process.

Table 5. Punctuation Removal Process

Normalization Sentences	Punctuation Removal
"Ini bagaimana sih aplikasinya, sudah tahu banyak yang memberi saran, bukan diperbaiki, malah dibiarkan seperti itu. Malah kebelet minta langganan 1 tahun lagi, yang perbulan saja seperti ini. Bagaimana sih managementnya, tidak lihat apa, jangan untungnya saja yang dikelola."	"ini bagaimana sih aplikasinya sudah tahu banyak yang memberi saran bukan diperbaiki malah dibiarkan seperti itu malah kebelet minta langganan 1 tahun lagi yang perbulan saja seperti ini bagaimana sih managementnya tidak lihat apa jangan untungnya saja yang dikelola"

Source : (Research Result, 2025)

Stop Words Removal

Common words that do not carry much specific meaning (stop words) are removed. Examples of common Indonesian stop words are : "ini", "yang", "di", "dan", "sudah", "bagaimana", "sih", "banyak", "bukan", "malah", "seperti", "itu", "minta", "lagi", "saja", "tidak", "apa", "jangan". The Stop Word Removal process can be seen in table 6.

Table 6. Stop Word Removal Process

Normalization Sentences	Stop Words Removal
"Ini bagaimana sih aplikasinya, sudah tahu banyak yang memberi saran, bukan diperbaiki, malah dibiarkan seperti itu. Malah kebelet minta langganan 1 tahun lagi, yang perbulan saja seperti ini. Bagaimana sih managementnya, tidak lihat apa, jangan untungnya saja yang dikelola."	"aplikasinya", "tahu", "memberi", "saran", "diperbaiki", "dibiarkan", "kebelet", "langganan", "1", "tahun", "perbulan", "managementnya", "lihat", "untungnya", "dikelola"

Source : (Research Result, 2025)

JITK (JURNAL ILMU PENGETAHUAN DAN TEKNOLOGI KOMPUTER)
VOL. 11. NO. 3 FEBRUARI 2026
P-ISSN: 2685-8223 | E-ISSN: 2527-4864
DOI: 10.33480/jitk.v11i3.7488

Stemming / Lemmatization

Each word from stop word removal is changed to its base form (stem or lemma). This process can be seen in table 7.

Table 7. Stemming Process

Stop Words Removal	Stemming/Lemmatization
"aplikasinya", "tahu", "memberi", "saran", "diperbaiki", "dibiarkan", "kebelet", "langganan", "1", "tahun", "perbulan", "managementnya", "lihat", "untungnya", "dikelola"	"aplikasi", "tahu", "beri", "saran", "baik", "biar", "kebelet", "langgan", "1", "tahun", "bulan", "manajemen", "lihat", "untung", "kelola"

Source : (Research Result, 2025)

Labeling

The labeling, or grouping, process can be done manually based on domain and language understanding, or automatically using a lexicon-based system. These systems work with a 'Lexicon Dictionary', a reference dictionary containing a collection of known sentiment words. The main function of this dictionary is to classify words, distinguishing between those that contain opinions and those that do not, with each word often having a predetermined weight. Therefore, labeling really requires this kind of sentiment dictionary. The sentiment labels were assigned manually based on an Indonesian sentiment lexicon adapted for app review contexts. Two annotators participated in the labeling, and inter-annotator agreement was measured using Cohen's Kappa to ensure reliability

Table 8. Labeling Example

Stemming / Lemmatization	Label
"aplikasi", "tahu", "beri", "saran", "baik", "biar", "kebelet", "langgan", "1", "tahun", "bulan", "manajemen", "lihat", "untung", "kelola"	*Negative*

Source : (Research Result, 2025)

Path 1: Feature Extraction using BERT

BERT Pretrained

The Bidirectional Encoder Representations from Transformers (BERT) model was utilized as a contextual feature extractor. Specifically, the IndoBERT-base pretrained model from Hugging Face was used due to its suitability for Indonesian linguistic structure.

a. Input tokens were preprocessed using the WordPiece tokenizer.
b. The [CLS] token representation was extracted from the final encoder layer as the sentence-level embedding.
c. Fine-tuning was performed for 3 epochs with a learning rate of 2e-5, and only the top transformer layers were unfrozen to prevent overfitting on the small dataset

Path 2: Feature Extraction with FastText and BLSTM

FastText Embedding

The FastText embedding model (pretrained on Indonesian Wikipedia) was used to generate subword-based word vectors. FastText's ability to represent out-of-vocabulary (OOV) words using character-level n-grams allows the model to handle morphological variations common in Indonesian.

The embeddings were fed into a Bidirectional Long Short-Term Memory (BLSTM) layer with 128 hidden units, which captures bidirectional dependencies in sequential text. A dropout rate of 0.3 was applied to prevent overfitting, and the Adam optimizer with a learning rate of 1e-4 was used for training.

Concatenation and Dense Layer

The outputs from both paths — contextual embedding from BERT ([CLS] token) and sequential embedding from BLSTM — were concatenated to form a unified feature representation. This step aims to combine global contextual understanding with local sequential dependencies.

The concatenated vector was then passed through:

a. Dense Layer 1: 64 neurons, *ReLU* activation
b. Dropout Layer: 0.3
c. Output Layer: 1 neuron, *Sigmoid* activation (for binary classification: positive vs. negative)

Model Training and Hyperparameter Tuning

The model was trained for 20 epochs using a batch size of 32, with early stopping based on validation loss to avoid overfitting. Hyperparameter optimization was performed using grid search over learning rate (1e-3 to 1e-5), dropout rate (0.2–0.4), and batch size (16–64). The best configuration was selected based on the highest validation accuracy.

RESULTS AND DISCUSSION

This section summarizes the results of the experiments conducted according to the previously planned research flow. This experiment focuses on analyzing text data from social media using a hybrid model of BLSTM FastText and BERT. Training and testing data are divided with an 80:20 division scheme. This study tests the ability of the combination of BLSTM FastText and BERT.

Model FastText BLSTM

Table 9 illustrates the configuration of these TP, FP, FN, and TN values in the confusion matrix of the combined FastText and BLSTM models, indicating a model that is Fairly Accurate Overall. Good at Finding Positive Cases (High Recall): The model does not miss many positive cases. Fairly Reliable at Predicting Positives (High Precision): Relative Errors Balanced: The FP (76) and FN (70) counts are not far apart, although FP is slightly higher. This means that the model is slightly more likely to misclassify negatives as positive than it is to miss positives. Slightly Lower Performance on Negative Class (Specificity): The model is slightly less effective at correctly identifying all negative cases compared to its ability to identify positive cases. Overall, the confusion matrix formed from these values depicts a model with solid performance, with a major strength in identifying positive cases (both in terms of recall and precision), but with room for improvement in reducing false positives to increase specificity.

Table 9. Confusion Matrix of BLSTM-FastText

		Actual Class	
		Class = Yes	Class = No
Predicted Class	Class = Yes	TP = 501	FP = 76
	Class = No	FN = 70	TN = 253

Source : (Research Result, 2025)

Model BERT

Table 10 describes the configuration of TP = 544, FP = 47, FN = 49, and TN = 260 values in the confusion matrix of the BERT model. These results indicate that the model has good overall performance. The model successfully classified 804 out of 900 data correctly, which reflects a fairly high level of reliability in predicting data in general. The model also showed good ability in finding positive cases, with 544 out of 593 positive cases successfully recognized, and only 49 positive cases escaped (FN). This shows that the model has a fairly strong detection ability for positive data, although there is still room to improve sensitivity to be more optimal. Furthermore, the model shows that the majority of positive predictions generated are correct, although there are 47 negative data that are misclassified as positive (FP). This means that the model is quite reliable in predicting positives, but the rate of misclassification of negatives as positive is slightly higher than a more precise model. The number of False Positive = 47 and False Negative = 49 is almost balanced, indicating that the model is not too biased towards one class, although there is a small tendency to misclassify negative data as positive. The TN value = 260 and FP = 47 indicate that the model is still relatively good at recognizing negative data, but the specificity level is slightly lower than the performance of detecting positive cases. This implies that the model can still be improved by reducing false positive predictions to strengthen performance in the negative class.

Table 10. Confusion Matrix of BERT

		Actual Class	
		Class = Yes	Class = No
Predicted Class	Class = Yes	TP = 544	FP = 47
	Class = No	FN = 49	TN = 260

Source : (Research Result, 2025)

Model FastText BLSTM BERT

Table 11 illustrates the configuration of TP = 598, FP = 25, FN = 27, and TN = 250 values in the combined confusion matrix of the FastText, BLSTM, and BERT models. These results indicate that the model has excellent overall performance, successfully classifying 848 out of 900 data points correctly, reflecting high reliability in predicting the data overall. Highly Reliable in Finding Positive Cases indicates that the model almost never misses data that are actually positive (only 27 out of 625 missed), indicating strong detection ability for positive cases. Furthermore, the model shows that most of its positive predictions are correct, or only a small number of negative data are misclassified as positive. The number of False Positives = 25 and False Negatives = 27 is almost balanced, with a fairly small value. This indicates that the model is not significantly biased towards one class, and only slightly biased towards being more easily misclassified as positive. The TN = 250 and FP value of only 25 indicate that this model is also quite effective in recognizing negative data. This model demonstrates very solid and balanced performance. It is well-suited for use in sentiment classification scenarios that prioritize accurate positive detection without sacrificing precision.

Table 11. Confusion Matrix of FastText BLSTM BERT

		Actual Class	
		Class = Yes	Class = No
Predicted Class	Class = Yes	TP = 598	FP = 25
	Class = No	FN = 27	TN = 250

Source : (Research Result, 2025)

Classification Performance Evaluation Value

Table 12 presents the comparative evaluation results of three different machine learning models or architectures, namely: "FastText-BLSTM", "BERT", and "FastText-BLSTM-BERT". The purpose of this table is to measure and compare the performance of each model in performing a specific task (most likely text

VOL. 11. NO. 3 FEBRUARI 2026
P-ISSN: 2685-8223 | E-ISSN: 2527-4864
DOI: 10.33480/jitk.v11i3.7488

classification or other natural language processing tasks) based on four industry-standard evaluation metrics. These metrics are Accuracy, Precision, Recall, and F1-score.

Table 12. Classification Performance Evaluation Values

Evaluation	FastText-BLSTM	BERT	FastText-BLSTM-BERT
Accuracy	83.77%	89.33%	94.22%
Precission	86.82%	92.04%	95.98%
Recall	87.74%	91.73%	95.68%
F1-score	87.28%	91.89%	95.83%

Source : (Research Result, 2025)

FastText-BLSTM Model

This model is a combination of FastText (for word representation) and BLSTM (Bidirectional Long Short-Term Memory, a type of recurrent neural network). Accuracy: 83.78% Overall, about 83.78% of all predictions made by the FastText-BLSTM model were correct. This is a measure of how often the model gives the correct answer from all the data it was tested on. Precision: 86.83%). When the FastText-BLSTM model predicted a data as "positive" (or the intended class), it was correct about 86.83% of the time. In other words, of all the data that the model claimed were positive, 86.83% of them were actually positive. The rest (about 13.17%) were false positives (mistaking negatives for positives). Recall: 87.74%. Of all the data that were actually "positive" in the dataset, the FastText-BLSTM model correctly identified about 87.74% of them. This means that the model is able to "remember" or find most of the positive cases. The rest (around 12.26%) are false negatives (missing positive cases and considering them negative). F1-score: 87.28% shows a balance between Precision and Recall for the FastText-BLSTM model. This value is the harmonic mean of both metrics, giving a single picture of how well the model is at minimizing false positives and false negatives simultaneously.

BERT Model

BERT (Bidirectional Encoder Representations from Transformers) is a state-of-the-art pre-trained language model that is often used as a basis for various NLP tasks. Accuracy 89.33%. The BERT model was correct in 89.33% of all predictions it made. This shows a general performance improvement over FastText-BLSTM. Precision 92.05%. When the BERT model predicted a data as "positive", it was correct about 92.05% of the time. This means that the BERT model is more reliable than FastText-BLSTM when it claims something is positive, with a lower false positive rate. Recall 91.74% , Of all the data that were actually "positive", the BERT model managed to correctly identify about 91.74% of them. This shows that BERT is better at finding relevant positive cases than FastText-BLSTM. F1-score: 91.89%, With an F1-score of 91.89%, the BERT model shows a better balance between Precision and Recall than FastText-BLSTM. This means that the model is more effective overall in managing the trade-off between not misclassifying negatives as positives and finding all positive cases.

***FastText-BLSTM-BERT Model* :**

The model appears to be a combination or ensemble of the three architectures, likely taking advantage of the strengths of each. Accuracy 94.22%: The FastText-BLSTM-BERT model was the most accurate, with 94.22% of its predictions correct. This is a significant improvement over the other two models, indicating the best overall performance. Precision 95.99%: When the FastText-BLSTM-BERT model predicted a data point as "positive," it was very accurate, correct about 95.99% of the time. This is the highest precision among the three models, meaning it produced the fewest false positives. Recall: 95.68% Of all the data points that were actually "positive," the FastText-BLSTM-BERT model correctly identified about 95.68% of them. This is the highest recall, indicating the best ability to find almost all positive cases. F1-score: 95.83% The highest F1-score of 95.83% indicates that the FastText-BLSTM-BERT model achieved the best balance between Precision and Recall. This indicates that this model is very effective in maximizing the identification of positive cases while minimizing misclassification.

In summary, the hybrid FastText-BLSTM-BERT model performs better because it combines FastText's subword-level generalization, BLSTM's sequential learning, and BERT's contextual representation. This synergy enhances the model's robustness against linguistic variations, data sparsity, and contextual ambiguity — key challenges in sentiment analysis on social media text. The results confirm that integrating multiple embedding and learning architectures can substantially improve model accuracy and generalization compared to single models.

Table 13 Summary of Technical Configuration

Component	Configuration
Embedding	FastText + BERT (IndoBERT-base)
BLSTM Units	128
Dense Layer	64 neurons, ReLU
Optimizer	Adam
Learning Rate	1e-4 (BLSTM) / 2e-5 (BERT fine-tuning)
Dropout	0.3
Batch Size	32

Component	Configuration
Epochs	20
Validation Split	0.2
Early Stopping	Enabled (patience = 3)
Class Weighting	Enabled
Loss Function	Binary Cross-Entropy

Table 13 *Summary of Technical Configuration* presents the detailed setup used in training the proposed hybrid FastText–BLSTM–BERT model. The embedding layer integrates FastText and IndoBERT-base, where FastText captures subword-level semantics and BERT contributes contextualized representations from large-scale pretraining. The BLSTM layer, consisting of 128 hidden units, enables the model to process text bidirectionally, capturing both forward and backward dependencies in the sequence.

A dense layer with 64 neurons and ReLU activation functions as a nonlinear transformation unit that combines and refines the concatenated features from both embedding paths. The Adam optimizer is employed due to its adaptive learning rate capability, facilitating stable and efficient convergence. To ensure optimal learning for both submodels, the learning rate is differentiated: 1e-4 for the BLSTM and 2e-5 for the fine-tuning of BERT layers.

To mitigate overfitting, a dropout rate of 0.3 is applied after each major layer, and early stopping is enabled with a patience value of 3 epochs to halt training when validation performance stagnates. Training is conducted using a batch size of 32 for 20 epochs, with 20% of the data reserved for validation. Furthermore, class weighting is implemented in the loss function to handle dataset imbalance between positive and negative classes. The binary cross-entropy loss function is adopted to optimize binary sentiment classification. This configuration ensures a balanced trade-off between model complexity, training stability, and generalization, particularly given the relatively limited dataset size used in this study.

CONCLUSION

This study successfully developed a FastText–BLSTM–BERT hybrid model for sentiment classification in Netflix app reviews. Evaluation results show that the hybrid model delivers superior performance compared to single models, as evidenced by significant improvements in accuracy, precision, recall, and F1-score. The integration of BERT, which effectively captures the global and contextual meaning of text, with FastText–BLSTM, which excels in processing subword semantics and sequential dependencies, produces a richer and more comprehensive feature representation. Achieving an accuracy of 94.22%, the proposed model demonstrates strong potential for application in various real-world sentiment analysis scenarios, such as monitoring customer satisfaction, analyzing public opinion on government policies, and enhancing digital marketing strategies on social media platforms.
Beyond its technical contributions, the findings imply that hybrid deep learning architectures combining contextual and sequential embeddings can substantially improve sentiment analysis performance in low-resource languages such as Indonesian. For future research, it is recommended to employ larger and more domain-diverse datasets, incorporate attention-based mechanisms or transformer variants for enhanced interpretability, and perform hyperparameter optimization and transfer learning to further increase model generalization and adaptability across different linguistic contexts.

REFERENCE

[1] Y. Bilan, O. Oliinyk, H. Mishchuk, and M. Skare, “Impact of information and communications technology on the development and use of knowledge,” *Technol. Forecast. Soc. Change*, vol. 191, no. March, p. 122519, 2023, doi: 10.1016/j.techfore.2023.122519.

[2] R. Lozano-Blasco, M. Mira-Aladrén., and M. Gil-Lamata, “Social media influence on young people and children: Analysis on Instagram, Twitter and YouTube,” *Comunicar*, vol. 30, no. 74, pp. 117–128, 2023, doi: http://doi.org/10.3916/C74-2023-10.

[3] T. Mantoro, M. A. Permana, and M. A. Ayu, “Crime index based on text mining on social media using multi classifier neural-net algorithm,” *Telkomnika (Telecommunication Comput. Electron. Control.*, vol. 20, no. 3, pp. 570–579, 2022, doi: https://doi.org/10.12928/TELKOMNIKA.v20i3.23321.

[4] S. Assegaff, E. Rasywir, and Y. Pratama, “Experimental of vectorizer and classifier for scrapped social media data,” *Telkomnika (Telecommunication Comput. Electron. Control.*, vol. 21, no. 4, pp. 815–824, 2023, doi: https://doi.org/10.12928/TELKOMNIKA.v21i4.24180.

[5] J. R. Hanaysha, “Impact of social media marketing features on consumer’s purchase decision in the fast-food industry: Brand trust as a mediator,” *Int. J. Inf. Manag. Data Insights*, vol. 2, no. 2, p. 100102, 2022, doi:

VOL. 11. NO. 3 FEBRUARI 2026
P-ISSN: 2685-8223 | E-ISSN: 2527-4864
DOI: 10.33480/jitk.v11i3.7488

10.1016/j.jjimei.2022.100102.

[6] H. R. Alhakiem and E. B. Setiawan, "Aspect-Bas1ed Sentiment Analysis on Twitter Using Logistic Regression with FastText Feature Expansion," *J. RESTI (Rekayasa Sist. dan Teknol. Informasi)*, vol. 6, no. 5, pp. 840–846, 2022, doi: https://doi.org/10.29207/resti.v6i5.4429.

[7] J. Jasmir, D. Z. Abidin, F. Fachruddin, and W. Riyadi, "Experimental of information gain and AdaBoost feature for machine learning classifier in media social data," *Indones. J. Electr. Eng. Comput. Sci.*, vol. 36, no. 2, pp. 1172–1181, 2024, doi: 10.11591/ijeecs.v36.i2.pp1172-1181.

[8] Z. A. Khan *et al.*, "Developing Lexicons for Enhanced Sentiment Analysis in Software Engineering: An Innovative Multilingual Approach for Social Media Reviews," *Comput. Mater. Contin.*, vol. 79, no. 2, pp. 2771–2793, 2024, doi: https://doi.org/10.32604/cmc.2024.046897.

[9] Y. Li, "Deep Learning-Based Natural Language Processing Methods for Sentiment Analysis in Social Networks," *Math. Probl. Eng.*, vol. 2022, 2022, doi: 10.1155/2022/1390672.

[10] P. Ray and A. Chakrabarti, "A Mixed approach of Deep Learning method and Rule-Based method to improve Aspect Level Sentiment Analysis," *Appl. Comput. Informatics*, vol. 18, no. 1–2, pp. 163–178, 2022, doi: 10.1016/j.aci.2019.02.002.

[11] W. Guo and A. Caliskan, "Detecting Emergent Intersectional Biases: Contextualized Word Embeddings Contain a Distribution of Human-like Biases," *AIES 2021 - Proc. 2021 AAAI/ACM Conf. AI, Ethics, Soc.*, pp. 122–133, 2021, doi: 10.1145/3461702.3462536.

[12] Z. Zhuang, Z. Liang, Y. Rao, H. Xie, and F. L. Wang, "Out-of-vocabulary word embedding learning based on reading comprehension mechanism," *Nat. Lang. Process. J.*, vol. 5, no. August, p. 100038, 2023, doi: 10.1016/j.nlp.2023.100038.

[13] J. Jasmir, W. Riyadi, S. R. Agustini, Y. Arvita, D. Meisak, and L. Aryani, "Bidirectional Long Short-Term Memory and Word Embedding Feature for," *J. RESTI (Rekayasa Sist. Dan Teknol. Informasi)*, vol. 6, no. 4, pp. 505–510, 2022, doi: https://doi.org/10.29207/resti.v6i4.4005.

[14] M. A. Hussain Sujon and H. Mustafa, "Comparative Study of Machine Learning Models on Multiple Breast Cancer Datasets," *Int. J. Adv. Sci. Comput. Eng.*, vol. 5, no. 1, pp. 15–24, 2023, doi: 10.62527/ijasce.5.1.105.

[15] S. Rapacz, P. Chołda, and M. Natkaniec, "A method for fast selection of machine-learning classifiers for spam filtering," *Electron.*, vol. 10, no. 17, 2021, doi: https://doi.org/10.3390/electronics10172083.

[16] F. N. N. H. Rossi Passarella, Siti Nurmaini, Muhammad Naufal Rachmatullah, Harumi Veny, "Development of a machine learning model for predicting abnormalities of commercial airplanes," *Res. Pract. Thromb. Haemost.*, p. 100137, 2023, doi: https://doi.org/10.1016/j.jsamd.2023.100613.

[17] I. A. Wisky, S. Defit, and G. W. Nurcahyo, "Development of Extraction Features for Detecting Adolescent Personality with Machine Learning Algorithms," *Int. J. INFORMATICS Vis.*, vol. 8, no. November, 2024, [Online]. Available: www.joiv.org/index.php/joiv

[18] H. Santoso, G. F. Soares, and C. M. Angelo, "Convolutional Neural Network - Based Recognition of Children ' s Facial Expressions in Response to Gaming," *Int. J. Adv. Sci. Comput. Eng.*, vol. 6, no. 1, pp. 152–157, 2024.

[19] N. Ansar, M. S. Ansari, M. Sharique, A. Khatoon, M. A. Malik, and M. M. Siddiqui, "A Cutting-Edge Deep Learning Method For Enhancing IoT Security," *Int. J. Adv. Sci. Comput. Eng.*, vol. 6, no. December, pp. 98–103, 2024, [Online]. Available: http://arxiv.org/abs/2406.12400

[20] A. H. Mohammed and A. H. Ali, "Survey of BERT (Bidirectional Encoder Representation Transformer) types," *J. Phys. Conf. Ser.*, vol. 1963, no. 1, 2021, doi: 10.1088/1742-6596/1963/1/012173.

[21] N. M. Gardazi, A. Daud, M. K. Malik, A. Bukhari, T. Alsahfi, and B. Alshemaimri, "BERT applications in natural language processing: a review," *Artif. Intell. Rev.*, vol. 58, no. 6, 2025, doi: 10.1007/s10462-025-11162-5.

[22] Y. Hao, L. Dong, F. Wei, and K. Xu, "Self-Attention Attribution: Interpreting Information Interactions Inside Transformer," *35th AAAI Conf. Artif. Intell. AAAI 2021*, vol. 14B, pp. 12963–12971, 2021, doi: 10.1609/aaai.v35i14.17533.

[23] E. Jasmir, Jasmir; Rasywir, H. Yani, and A. Nugroho, "Comparison of word embedding features using deep learning in sentiment analysis," *TELKOMNIKA Telecommun. Comput. Electron. Control*, vol. 23, no. 2, pp.

416–425, 2025, doi: 10.12928/TELKOMNIKA.v23i2.26223.

[24] A. Mahmoud and M. Zrigui, "BLSTM-API: Bi-LSTM Recurrent Neural Network-Based Approach for Arabic Paraphrase Identification," *Arab. J. Sci. Eng.*, vol. 46, no. 4, pp. 4163–4174, 2021, doi: 10.1007/s13369-020-05320-w.

[25] M. Umer *et al.*, "Impact of convolutional neural network and FastText embedding on text classification," *Multimed. Tools Appl.*, vol. 82, no. 4, pp. 5569–5585, 2023, doi: 10.1007/s11042-022-13459-x.

[26] E. Hashmi, S. Y. Yayilgan, M. M. Yamin, S. Ali, and M. Abomhara, "Advancing Fake News Detection: Hybrid Deep Learning With FastText and Explainable AI," *IEEE Access*, vol. 12, no. March, pp. 44462–44480, 2024, doi: 10.1109/ACCESS.2024.3381038.

[27] F. S. Utomo, "Information Retrieval Method for the Qur ' an based on FastText and Latent Semantic Indexing," *J. Sist. Inf.*, vol. 14, pp. 1014–1024, 2025.

[28] N. Haffar and M. Zrigui, "A Synergistic Bidirectional LSTM and N-gram Multi-channel CNN Approach Based on BERT and FastText for Arabic Event Identification," *ACM Trans. Asian Low-Resource Lang. Inf. Process.*, vol. 22, no. 11, 2023, doi: 10.1145/3626568.

[29] N. Haffar and M. Zrigui, "BERT embedding model for arabic temporal relation classification using hybrid deep learning architectures and linguistic features," *Int. J. Knowledge-Based Intell. Eng. Syst.*, vol. 29, no. 1, pp. 32–54, 2025, doi: 10.3233/KES-230066.

[30] H. Gunduz, "Comparative analysis of BERT and FastText representations on crowdfunding campaign success prediction," *PeerJ Comput. Sci.*, vol. 10, 2024, doi: 10.7717/PEERJ-CS.2316.

[31] S. Shreyashree, P. Sunagar, S. Rajarajeswari, and A. Kanavalli, "BERT-Based Hybrid RNN Model for Multi-class Text Classification to Study the Effect of Pre-trained Word Embeddings," *Int. J. Adv. Comput. Sci. Appl.*, vol. 13, no. 9, pp. 667–674, 2022, doi: 10.14569/IJACSA.2022.0130979.

[32] N. K, K. M, S. V. Easwaramoorthy, D. C R, S. Yoo, and J. Cho, "Hybrid approach of deep feature extraction using BERT– OPCNN & FIAC with customized Bi-LSTM for rumor text classification," *Alexandria Eng. J.*, vol. 90, no. October 2023, pp. 65–75, 2024, doi: 10.1016/j.aej.2024.01.056.