Available online at http://scik.org
Commun. Math. Biol. Neurosci. 2025, 2025:16
https://doi.org/10.28919/cmbn/9006
ISSN: 2052-2541

EMOTION CLASSIFICATION IN SOCIAL MEDIA POSTS RELATED TO TELECOMMUNICATION SERVICES USING BIDIRECTIONAL LSTM

SANDRINA NAJWA[1], SRI WINARNI[2], ANINDYA APRILIYANTI PRAVITASARI[2,*], YUYUN HIDAYAT[2]

[1]Master's Program in Applied Statistics, Department of Statistics, Padjadjaran University, Indonesia

[2]Department of Statistics, Padjadjaran University, Indonesia

Abstract: Social media has become a vital platform for sharing opinions, with 139 million users in Indonesia as of January 2024. Telecommunications companies can leverage feedback from platforms like Twitter and Instagram to understand customer sentiment and improve service quality. This study applies Bidirectional Long Short-Term Memory (BiLSTM) and FastText word embeddings to classify emotions in social media posts related to major Indonesian telecom providers, including Telkomsel, Indosat, XL, and Axis. Using Borderline Synthetic Minority Oversampling Technique (B-SMOTE) to address class imbalance, the model categorizes six basic emotions: happiness, sadness, fear, anger, disgust, and surprise. The optimal model, trained over 18 epochs, includes 64 BiLSTM units, 128 dense layer neurons, a 0.3 dropout rate, a batch size of 32, and a learning rate of 0.001. It achieved 93.51% accuracy and a 93.48% F1 score on unseen data, demonstrating strong performance in predicting customer emotions. This approach provides valuable insights for improving customer engagement and service in the telecommunications industry.

Keywords: emotion classification; BiLSTM; social media; telecommunications.

2020 AMS Subject Classification: 68T50.

*Corresponding author

E-mail address: anindya.apriliyanti@unpad.ac.id

Received November 05, 2024

1. INTRODUCTION

Currently, the internet is predominantly used for accessing social media. Social media platforms enable individuals to share information without the need for direct interaction, providing each person with the freedom to express their opinions [1]. As of January 2024, it is noteworthy that internet users in Indonesia have reached 185.3 million, with the internet penetration rate increasing to 66.5%. A significant portion of this internet usage is dedicated to accessing social media, with 139 million users, or approximately 49.9% of the total population in Indonesia, using social media platforms as of January 2024 [2].

The widespread use of social media not only influences the way society interacts but also provides opportunities for companies, especially those in the telecommunications sector, to gain a deeper understanding of customer needs and perceptions [3]. Through platforms such as Twitter, Facebook, and Instagram, consumers can easily convey their feedback, suggestions, or complaints regarding the services they receive [4]. For telecommunications companies, feedback from social media serves as a valuable source of information to evaluate service quality, enhance customer satisfaction, and identify issues that require immediate attention [5]. In this context, the ability to classify emotions in social media conversations becomes crucial, particularly in understanding customer sentiments and providing more timely and targeted responses. According to Paul Ekman, human emotions can be categorized into six fundamental emotions: happiness, sadness, fear, anger, disgust, and surprise [6]. Each of these emotions plays a significant role in understanding consumers' reactions to the services provided by companies, especially in the telecommunications sector. For instance, happiness often reflects satisfaction with a service, whereas anger typically indicates dissatisfaction or complaints.

Unstructured text data, such as social media posts, presents complex challenges for text classification, as it requires attention to several important aspects, including sequential data, semantics, and syntactic. Sequential data refers to the ordered nature of text, where the sequence of words or phrases affects the overall meaning [7]. Each word is processed in sequence to recognize patterns in sentences, making it crucial to consider word order for accurate message comprehension [8]. The semantic aspect demands an understanding of the meaning of words and phrases in a particular context, which often varies in Natural Language Processing (NLP). Meanwhile, the syntactic aspect relates to the arrangement of grammatical elements such as subject, verb, and object, which help in understanding relationships between words in a sentence.

This study aims to employ the Bidirectional Long Short-Term Memory (BiLSTM) method as a deep learning approach to classify six emotions: happiness, sadness, fear, anger, disgust, and surprise in telecommunications-related discussions on social media. The BiLSTM method is recognized for its ability to handle unstructured data, sequential data, as well as the semantic and syntactic aspects essential in text analysis [9]–[11]. Additionally, BiLSTM is capable of processing words in both directions simultaneously, i.e., forward (from the beginning to the end of the text) and backward (from the end to the beginning), allowing it to capture richer and more complex contexts and better understand the relationships between words in a text [12].

2. MATERIALS AND METHODS

The dataset for this investigation is derived from customer comments discussing telecommunications, specifically mentions and keywords related to Indosat, Telkomsel, XL, and Axis. These data were collected from various social media platforms, including Twitter, YouTube, Instagram, Facebook, and TikTok, over the observation period from June 1, 2023, to August 11, 2023, resulting in a total of 11,730 posts. Data collection was facilitated using the Ripple10 dashboard, which is only accessible to certain companies. The labeling process to classify these posts into six emotional categories "happiness", "surprise", "sadness", "fear", "anger", and "disgust" was conducted manually. Posts expressing fear were assigned a label of (1), disgust (2), anger (3), surprise (4), happiness (5), and sadness (6). The flowchart attached to Figure 1 outlines the research methodology employed in conducting this study.

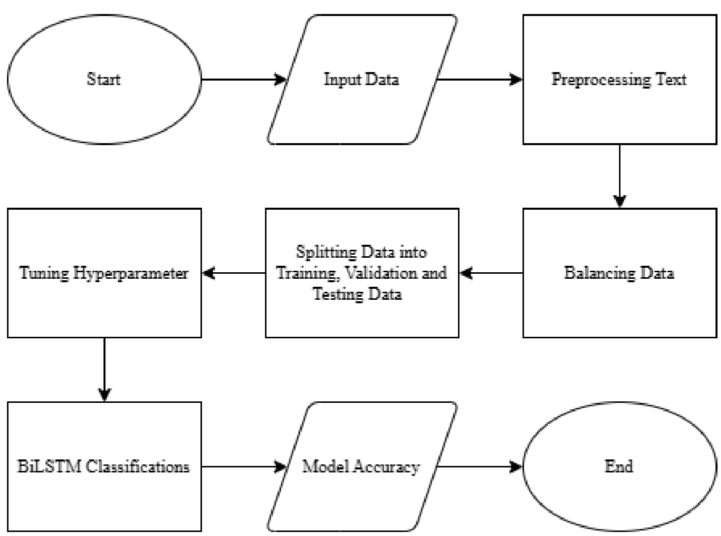


Figure 1. Research process diagram

2.1 Text Preprocessing

The initial stage in text mining is text preprocessing [13]. Text preprocessing is a crucial step that must be undertaken to ensure that the data can be optimally processed by the text mining system [14]. The purpose of text preprocessing is to convert unstructured text data into a more structured form [15]. This process involves several steps, including cleaning, where irrelevant characters such as hashtags, URLs, mentions, and symbols are removed to reduce noise [16]. Next, case folding is applied to convert all characters to lowercase, which aids in standardizing text and improving classification accuracy [17]. Tokenization breaks down the text into its smallest units, i.e., words, in preparation for further analysis [18]. Spelling normalization is performed to correct misspellings or replace certain abbreviations in the text data, while stopword removal eliminates less informative words, allowing for a focus on more meaningful terms [17]. Stemming is a linguistic process that removes affixes from words, grouping words with the same root, and improving the retrieval of documents in search and indexing systems [19]. To proceed with text classification, natural and unstructured text must be transformed into numerical data [20]. The next step involves word embedding, where words are mapped into vectors to capture semantic and syntactic meaning. FastText was chosen as the method for this study due to its superior performance in text classification [21]. Figure 2 illustrates the sequential steps involved in the text preprocessing phase, starting from input to word embedding, which ensures that the text data is prepared effectively for further analysis and classification

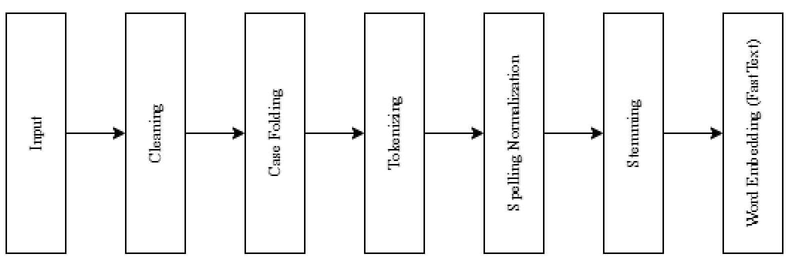


Figure 2. Text preprocessing steps

2.2 Borderline Synthetic Minority Oversampling Technique (B-SMOTE)

Borderline-SMOTE, a variation of the SMOTE method proposed by Han, WenYuan, and Bing-Huan in 2005, is an oversampling technique focused on minority class samples that lie on the border of the majority class [22]. This focus on borderline samples is due to the higher risk of misclassification, as these samples are most vulnerable to incorrect predictions [23]. Therefore, it is crucial to estimate an optimal decision boundary to improve classification accuracy in the context of imbalanced data. Borderline-SMOTE also effectively prevents the creation of irrelevant synthetic data, which can degrade model performance if oversampling is done randomly without considering the context of the class boundaries.

2.3 Bidirectional Long Short-Term Memory (BiLSTM)

Long Short-Term Memory (LSTM) is an improvement of the Recurrent Neural Network (RNN) architecture, incorporating memory cells to store long-term information, which helps overcome the vanishing gradient problem in RNNs when processing long sequential data [24]. This method has been widely implemented in various fields, such as speech recognition, natural language processing, and sequence modeling [25]. Figure 3 illustrates the architecture of a single LSTM neuron [26].

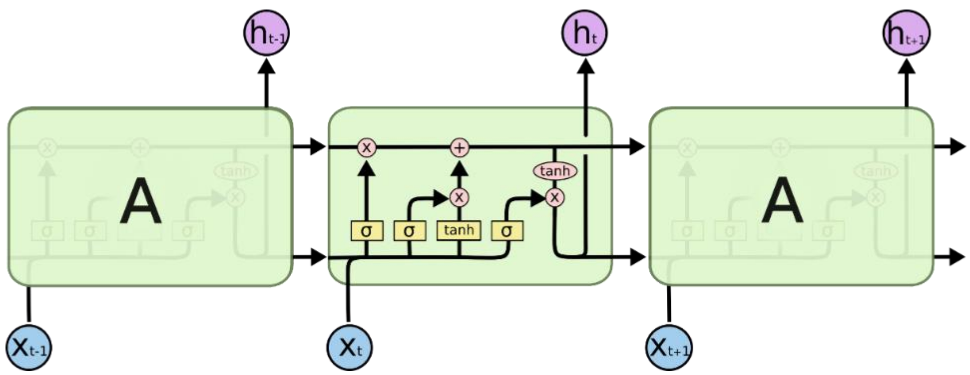


Figure 3. Single neuron within the LSTM architecture

LSTM consists of four primary gates, namely the forget gate, input gate, cell state, and output gate. The forget gate takes in the inputs $\boldsymbol{h}_{t-1}$ and $\boldsymbol{x}_t$ to generate a value between 0 and 1 for $\boldsymbol{c}_{t-1}$. If the forget gate value is close to 1, the cell state information will be retained, whereas if it is near 0, the information is discarded. The input gate is the second gate that determines what new information will be added to the cell state through a combination of the sigmoid and tanh functions. The sigmoid function decides which information will be updated, and the tanh function creates a new update $\tilde{\boldsymbol{c}}_t$ to be added to the cell state. The output gate is the final gate that produces the hidden state $\boldsymbol{h}_t$ based on the updated cell state. The LSTM equations are represented with the following notations [27].

$$\boldsymbol{f}_t = \sigma(\boldsymbol{W}_{xf}.\boldsymbol{x}_t + \boldsymbol{W}_{hf}.\boldsymbol{h}_{t-1} + \boldsymbol{b}_f) \tag{1}$$

$$\boldsymbol{i}_t = \sigma(\boldsymbol{W}_{xi}.\boldsymbol{x}_t + \boldsymbol{W}_{hi}.\boldsymbol{h}_{t-1} + \boldsymbol{b}_i) \tag{2}$$

$$\tilde{\boldsymbol{c}}_t = \tanh\,(\boldsymbol{W}_{xc}.\boldsymbol{x}_t + \boldsymbol{W}_{hc}.\boldsymbol{h}_{t-1} + \boldsymbol{b}_c) \tag{3}$$

$$\boldsymbol{c}_t = \boldsymbol{f}_t \odot \boldsymbol{c}_{t-1} + \boldsymbol{i}_t \odot \tilde{\boldsymbol{c}}_t \tag{4}$$

$$\boldsymbol{o}_t = \sigma(\boldsymbol{W}_{xo}.\boldsymbol{x}_t + \boldsymbol{W}_{ho}.\boldsymbol{h}_{t-1} + \boldsymbol{b}_0) \tag{5}$$

$$\boldsymbol{h}_t = \boldsymbol{O}_t \odot tanh\ (\boldsymbol{c}_t) \tag{6}$$

where $\boldsymbol{W}_{xf}, \boldsymbol{W}_{xi}, \boldsymbol{W}_{xc}, \boldsymbol{W}_{xo}$ represent the trained weights, $\boldsymbol{b}_f, \boldsymbol{b}_{i,}, \boldsymbol{b}_c, \boldsymbol{b}_0$ are the trained biases, σ denotes the sigmoid activation function, $\boldsymbol{x}_t$ is the input at time *t*, $\boldsymbol{h}_{t-1}$ is hidden state from previous time step, $\boldsymbol{c}_t$ represent the cell state at time *t*, $\boldsymbol{h}_t$ is output at time *t*, $\mathbf{f}_t$ is the forget gate at time *t*, $\boldsymbol{i}_t$ is the input gate at time *t*, $\boldsymbol{o}_t$ is the output gate at time *t* and $\odot$ indicates the element-wise product.

However, LSTM processes sequences in only one direction (forward), which limits its ability to fully capture the context of longer sequences. Bidirectional Long Short-Term Memory (BiLSTM) addresses this issue by processing information in both directions (forward and backward), enabling simultaneous modeling of the context from both preceding and following words [28].

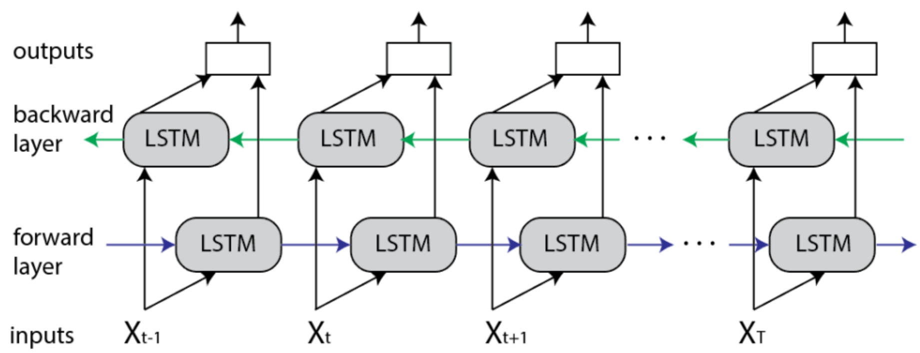


Figure 4. Bidirectional LSTM architecture

As shown in Figure 4, the BiLSTM architecture consists of two layers: a forward layer that processes sequences from the beginning to the end, and a backward layer that processes them from the end to the beginning [29]. The two hidden states, $\boldsymbol{h}_t^{forward}$ dan $\boldsymbol{h}_t^{backward}$ from the LSTM are integrated to produce the final hidden state, $\boldsymbol{h}_t^{BiLSTM}$ as outlined in Equation (7) [30].

$$\boldsymbol{h}_t^{BiLSTM} = \boldsymbol{h}_t^{forward} \oplus \boldsymbol{h}_t^{backward} \tag{7}$$

with this bidirectional architecture, BiLSTM enables better contextual information capture and models longer dependencies in text classification [12].

2.4 Evaluation

The classification model's performance is measured using a confusion matrix. **[31]**. The confusion matrix compares the system's predictions with the actual classification results. Four key terms are used in this evaluation: True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN), which allow the calculation of performance metrics, including accuracy, precision, recall, and F1-score [32].

Table 1. Confusion Matrix 6×6

		Prediction					
		Class 1	Class 2	Class 3	Class 4	Class 5	Class 6
Actual	Class 1	x_{11}	x_{12}	x_{13}	x_{14}	x_{15}	x_{16}
	Class 2	x_{21}	x_{22}	x_{23}	x_{24}	x_{25}	x_{26}
	Class 3	x_{31}	x_{32}	x_{33}	x_{34}	x_{35}	x_{36}
	Class 4	x_{41}	x_{42}	x_{43}	x_{44}	x_{45}	x_{46}
	Class 5	x_{51}	x_{52}	x_{53}	x_{54}	x_{55}	x_{56}
	Class 6	x_{61}	x_{62}	x_{63}	x_{64}	x_{65}	x_{66}

Table 1 presents the 6×6 confusion matrix used in multiclass classification, comparing the model's predicted classification to the actual classification. From this confusion matrix, various evaluation metrics can be computed to assess the performance of the model. [33].

$$FN_i = \sum_{\substack{j=1 \\ j \neq i}}^{n} x_{ij} \tag{8}$$

$$FP_i = \sum_{\substack{j=1 \\ j \neq i}}^{n} x_{ji} \tag{9}$$

$$TN_i = \sum_{\substack{j=1 \\ j \neq i}}^{n} \sum_{\substack{k=1 \\ k \neq i}}^{n} x_{jk} \tag{10}$$

$$TP_{all} = \sum_{j=1}^{n} x_{jj} \tag{11}$$

using these values, the following key metrics such as precision (P_i), recall (R_i), F1-score (F_i) and accuracy (A) can be calculated:

$$P_i = \frac{TP_{all}}{TP_{all} + FP_i} \tag{12}$$

$$R_i = \frac{TP_{all}}{TP_{all} + FN_i} \tag{13}$$

$$F_i = \frac{2 * R * P}{R + P} \tag{14}$$

$$A = \frac{TP_{all}}{total\ \ testing\ data} \tag{15}$$

where $i, j, k = 1{,}2{,}3{,}4{,}5{,}6$ these evaluation metrics offer a thorough assessment of the model's overall performance.

3. Main Results

3.1 Text preprocessing

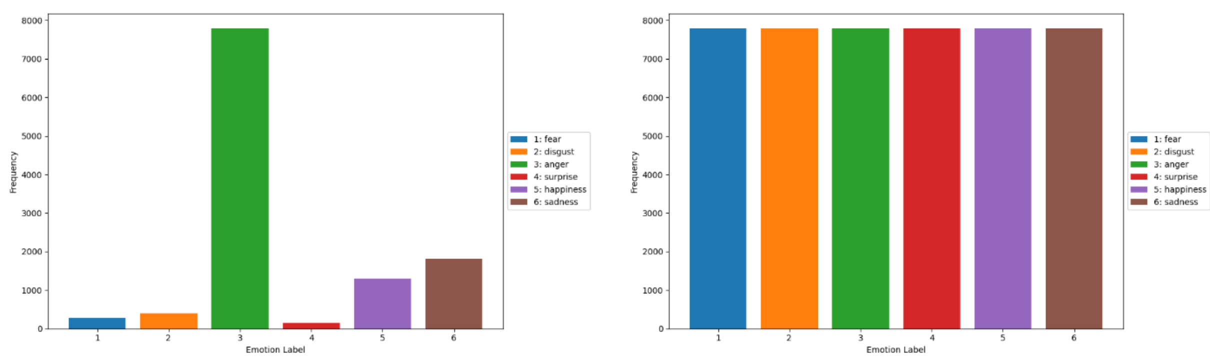
Text preprocessing is essential to ensure that text data can be properly interpreted and processed by text-processing systems. The steps involved in text preprocessing are outlined in Table 2 below. These steps include various processes, such as cleaning, which involves removing unwanted characters like hashtags, URLs, mentions, and symbols to retain the original data. Following this, case folding is applied to convert all characters to lowercase, enhancing classification accuracy and ensuring consistency across the dataset. The next step is tokenization, which breaks the text into individual words based on spaces, making it more readable for computational systems. Spelling normalization addresses spelling errors and replaces abbreviations with their full forms. Additionally, stopword removal eliminates non-essential words that do not significantly impact analysis, utilizing the NLTK library, while stemming reduces words to their root form by removing affixes, using the Sastrawi library. After that, the vocabulary tokens are converted into a sequence of numbers by assigning an integer to each token based on its position in the vocabulary index. Collectively, these preprocessing steps aim to refine and simplify the data, improve its quality, and prepare it for subsequent analysis.

Table 2. Text Preprocessing Result

Process	Result
Input	Duh ampun telkomsel ANJRIT gaada angin gada ujan sinyalnya ilang terus Ã°Å¸ËoeÂ
Cleaning	Duh ampun telkomsel ANJRIT gaada angin gada ujan sinyalnya ilang terus
Case Folding	duh ampun telkomsel anjrit gaada angin gada ujan sinyalnya ilang terus
Tokenizing	['duh', 'ampun', 'telkomsel', 'anjrit', 'gaada', 'angin', 'gada', 'ujan', 'sinyalnya', 'ilang', 'terus']
Spelling Normalization	['aduh', 'ampun', 'telkomsel', 'anjing', 'tidak ada', 'angin', 'tidak ada', 'hujan', 'sinyalnya', 'hilang', 'terus']
Stopword Removal	['aduh', 'ampun', 'telkomsel', 'anjing', 'tidak ada', 'angin', 'tidak ada', 'hujan', 'sinyalnya', 'hilang']
Stemming	['aduh', 'ampun', 'telkomsel', 'anjing', 'tidak ada', 'angin', 'tidak ada', 'hujan', 'sinyal', 'hilang']

3.2 Imbalanced Data

Imbalanced data occurs when there is a significant disproportion between class sizes in a dataset, with the majority class having more instances and the minority class having fewer. This imbalance can lead to biased learning, where the model becomes more proficient at predicting the majority class, thus compromising its ability to accurately classify minority instances. As a result, model performance declines, particularly in recognizing and classifying minority data. To address this issue, the B-SMOTE (Borderline Synthetic Minority Oversampling Technique) method is applied, which focuses on generating synthetic data for minority class instances that lie near the decision boundary, making them more prone to misclassification.



(a) **(b)**

Figure 5. Data balancing using B-SMOTE,

(a) Plot showing the distribution of data before B-SMOTE balancing is applied, (b) Plot showing the distribution of data after B-SMOTE balancing is applied

Figure 5 illustrates the data balancing process using the B-SMOTE method. In plot (a), we observe a significant class imbalance, with the 'anger' emotion (3) being the majority class, while the other emotions have considerably fewer instances. This imbalance can lead to a classification model that is biased toward the majority class, reducing its ability to accurately classify minority classes.

After applying B-SMOTE, as depicted in plot (b), the distribution across all emotion labels is balanced, with each class containing 7,786 instances. This balancing process aims to enhance the model’s ability to recognize and classify minority classes, thus improving overall model performance by ensuring fair and accurate predictions across all emotion categories.

3.3 Splitting Dataset

This stage involves dividing the research data split into 80% training data, 10% validation data, and 10% test data using the stratified hold out method to maintain balanced class distribution across each set. The sample size for each set is calculated proportionally based on the emotion classes.

The training data consists of 37,372 samples, with 6,231 for the fear class, 6,249 for disgust, 6,233 for anger, 6,181 for surprise, 6,279 for happiness, and 6,199 for sadness. The validation data comprises 4,672 samples, with 758 for fear, 771 for disgust, 807 for anger, 794 for surprise, 745 for happiness, and 797 for sadness. The testing data consists of 4,672 samples, with 797 for fear, 766 for disgust, 746 for anger, 811 for surprise, 762 for happiness, and 790 for sadness.

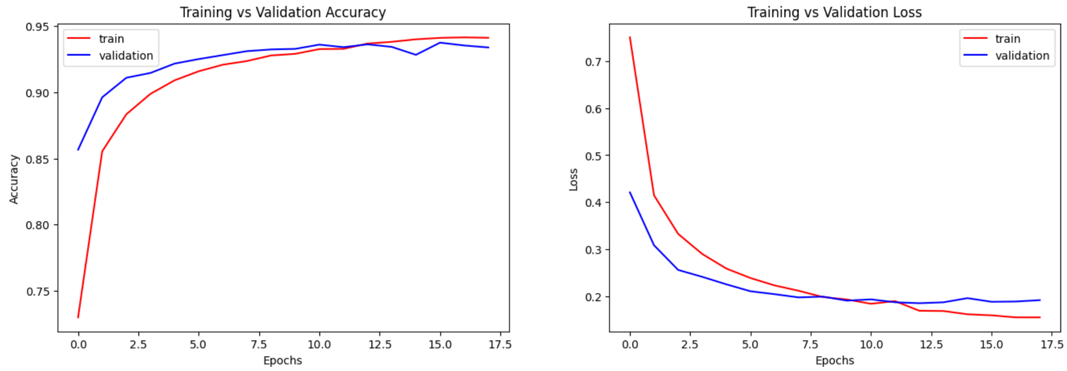
3.4 Implementing Bidirectional LSTM for Data Modeling

The modeling is structured with a network architecture consisting of four layers. First, the embedding layer with 300 dimensional vectors generated by the FastText model. Next, the data is passed to the BiLSTM layer, followed by a fully connected dense layer with dropout added to minimize overfitting. Finally, the output dense layer with 6 units and a softmax activation function is used to handle the multiclass classification task. To achieve the optimal model, precise hyperparameters are essential. The training data, which constitutes 80% of the dataset, is used for training, while 10% of the data is allocated for validation during training. Early Stopping with a maximum of 100 epochs is applied to prevent overfitting. The hyperparameters being tested include the number of neurons, dropout rate, batch size, and learning rate. Table 3 presents the results of the hyperparameter tuning process, outlining the BiLSTM parameters that were evaluated to identify the optimal configuration.

Table 3. Tuning hyperparameter

Parameter	Value	Parameter Optimal
BiLSTM Layer Neurons	32, 64, 128	64
Dense Layer Neurons	32, 64, 128	128
Dropout Rate	0.1, 0.2, 0.3, 0.4, 0.5	0.3
Optimizer	Adam	Adam
Batch Size	16, 32, 64, 128, 256	32
Learning Rate	0.01, 0.001, 0.0001	0.001
Epochs	100	18

Based on the hyperparameter tuning, the optimal parameters for the BiLSTM model were identified as 64 units for the BiLSTM layer, 128 neurons for the dense layer, a dropout rate of 0.3, a batch size of 32, and a learning rate of 0.001. Figure 6 shows the training and validation performance of the model using the BiLSTM method with these optimal parameters.



(a) **(b)**

Figure 6. Training and Validation Graphs,

(a) Accuracy Graph for Training and Validation, (b) Loss Graph for Training and Validation

As shown in Figure 6, the BiLSTM model achieved a training accuracy of 93.72% and a final loss value of 0.184. The validation accuracy exhibited a similar upward trend, indicating that the model effectively generalizes to unseen data. The continuous decrease in both training and validation loss further suggests effective learning, with minimal signs of overfitting throughout the training process.

3.5 Evaluation Model

The final step in this research is evaluating the model using the test data, consisting of 4,672 samples that the model has not previously seen. The confusion matrix in Table 4 shows that the model correctly predicted 4,369 out of 4,672 samples. The class with the highest misclassification rate is sadness, which was often predicted as anger, possibly due to ambiguous sentences containing both emotions. The use of Borderline SMOTE may have also contributed to misclassifications by altering the semantic structure of the data.

Table 4. Confusion matrix result

		Prediction					
		Fear	Disgust	Anger	Surprise	Happiness	Sadness
	Fear	772	0	21	0	4	0
	Disgust	0	744	20	0	2	0
	Anger	4	9	711	0	15	7
Actual	Surprise	1	0	6	803	1	0
	Happiness	0	0	20	0	742	0
	Sadness	6	2	163	2	20	597

The model's performance was evaluated using precision, recall, F1-score, and accuracy metrics derived from the confusion matrix in Table 4. These metrics were computed using Equation (12) for precision, Equation (13) for recall, Equation (14) for F1-score, and Equation (15) for accuracy. As shown in Table 5, the model achieved an overall F1-score of 93.48% and an accuracy of 93.51%. The sadness class had the lowest F1-score (82.58%), likely due to data ambiguity and the effect of oversampling. Despite this, the model demonstrated strong overall performance in classifying emotions in social media posts, particularly within discussions related to telecommunications topics.

Table 5. Performance matrix

	Precision	Recall	F1-Score	Accuracy
Fear	0.9860	0.9686	0.9772	
Disgust	0.9854	0.9713	0.9783	
Anger	0.7556	0.9531	0.8429	
Surprise	0.9975	0.9901	0.9938	0.9351
Happiness	0.9464	0.9738	0.9599	
Sadness	0.9403	0.7361	0.8258	
overall	0.9432	0.9354	0.9348	

4. CONCLUSIONS

Based on the results of the study, the application of the Bidirectional Long Short-Term Memory (BiLSTM) method demonstrated strong performance in classifying emotions on social media, specifically within discussions related to telecommunications topics. The best model was obtained with a configuration of 64 units in the BiLSTM layer, 128 neurons in the dense layer, a dropout rate of 0.3, the Adam optimizer, a batch size of 32, a learning rate of 0.001, and 18 epochs. On unseen data, the model achieved an accuracy of 93.51% and an overall F1-score of 93.48%. These results indicate that the model is effective for automatically predicting six fundamental emotions, which include fear, disgust, anger, surprise, happiness, and sadness in telecommunications-related discussion.

ACKNOWLEDGEMENT

The authors thank Ivonesia Solusi Data (Ivosights) for providing customer comments about prominent telecommunications providers in Indonesia, collected from Twitter, YouTube, Instagram, Facebook, and TikTok. The authors also express their gratitude to Padjadjaran University for funding this research.

Conflict of Interests

The authors declare that there is no conflict of interests.

References

[1] Y.P. Kommineni, S. Somanchi, B.S.P. Kumar, et al. Social Media Text to Formal Representation: A Knowledge Graph Approach, in: 2023 9th International Conference on Advanced Computing and Communication Systems, IEEE, Coimbatore, India, 2023: pp. 1691–1696. https://doi.org/10.1109/ICACCS57279.2023.10112722.

[2] DataReportal, Digital 2024: Indonesia, (2024). https://datareportal.com/reports/digital-2024-indonesia, Accessed Sep. 13, 2024.

[3] N. Almuhanna, Z.H. Alharbi, Factors Affecting Customer Satisfaction with The Telecommunication Industry in Saudi Arabia, TEM J. 12 (2023), 423–433. https://doi.org/10.18421/TEM121-52.

[4] K. Saranya, D. Sudharson, M. Aathmika, et al. Twitter Based Complaint Management System for Rail Transit, in: 2023 2nd International Conference on Advancements in Electrical, Electronics, Communication, Computing and Automation, Coimbatore, India, 2023: pp. 1–6. https://doi.org/10.1109/ICAECA56562.2023.10199383.

[5] Z. Sarmast, S. Shokouhyar, S.H. Ghanadpour, S. Shokoohyar, Unravelling the Potential of Social Media Data Analysis to Improve the Warranty Service Operation, Ind. Manag. Data Syst. 123 (2023), 1281–1309. https://doi.org/10.1108/IMDS-07-2022-0427.

[6] F.A.C. Salinas, G.G.R. Ortiz, P.C.M. Suarez, Recognition of Emotions through Facial Expression Analysis, S. Florida J. Dev. 2 (2021), 2102–2118. https://doi.org/10.46932/sfjdv2n2-076.

[7] I.L. Rais, J. Jondri, Klasifikasi Data Kuesioner dengan Metode Recurrent Neural Network, eProc. Eng. 7 (2020), 2817–2826.

[8] D.J.M. Pasaribu, K. Kusrini, S. Sudarmawan, Peningkatan Akurasi Klasifikasi Sentimen Ulasan Makanan Amazon Dengan Bidirectional LSTM Dan Bert Embedding, Inspiration: Jurnal Teknologi Informasi Dan Komunikasi 10 (2020), 9-20. https://doi.org/10.35585/inspir.v10i1.2568.

[9] L. Li, Emotion Analysis Method of Teaching Evaluation Texts Based on Deep Learning in Big Data Environment, Comput. Intell. Neurosci. 2022 (2022), 9909209. https://doi.org/10.1155/2022/9909209.

[10] S.N. Yu, S.W. Wang, Y.P. Chang, Improving Distinguishability of Photoplethysmography in Emotion Recognition Using Deep Convolutional Generative Adversarial Networks, IEEE Access 10 (2022), 119630–119640. https://doi.org/10.1109/ACCESS.2022.3221774.

[11] J. Yang, X. Huang, H. Wu, X. Yang, EEG-Based Emotion Classification Based on Bidirectional Long Short-Term Memory Network, Procedia Comput. Sci. 174 (2020), 491–504. https://doi.org/10.1016/j.procs.2020.06.117.

[12] G. Liu, J. Guo, Bidirectional LSTM with Attention Mechanism and Convolutional Layer for Text Classification, Neurocomputing 337 (2019), 325–338. https://doi.org/10.1016/j.neucom.2019.01.078.

[13] V.E. Jacob, A.S.M. Lumenta, A. Jacobus, Rancang Bangun Aplikasi Kemiripan Dokumen Dengan Sumber-Sumber Internet, J. Tek. Inform. 14 (2019), 160–161.

[14] A. Rajesh, T. Hiwarkar, Exploring Preprocessing Techniques for Natural LanguageText: A Comprehensive Study Using Python Code, Int. J. Eng. Technol. Manag. Sci. 7 (2023), 390–399. https://doi.org/10.46647/ijetms.2023.v07i05.047.

[15] V. Singh, V.K. Asari, S. Kumar, R.B. Patel, eds., Computational Methods and Data Engineering: Proceedings of ICMDE 2020, Springer Singapore, 2021. https://doi.org/10.1007/978-981-15-6876-3.

[16] M. Pota, M. Ventura, H. Fujita, et al. Multilingual Evaluation of Pre-Processing for BERT-Based Sentiment Analysis of Tweets, Expert Syst. Appl. 181 (2021), 115119. https://doi.org/10.1016/j.eswa.2021.115119.

[17] Y. HaCohen-Kerner, D. Miller, Y. Yigal, The Influence of Preprocessing on Text Classification Using a Bag-of-Words Representation, PLOS ONE 15 (2020), e0232525. https://doi.org/10.1371/journal.pone.0232525.

[18] P. Yannis, Automated Text Analysis: The Application of Automatic Text Processing in the Social Sciences, in: Research Methods in the Social Sciences: An A-Z of Key Concepts, Oxford University Press, 2021.

[19] W.B. Demilie, Implemented Stemming Algorithms for Information Retrieval Applications, J. Inf. Eng. Appl. 10 (2020), 1-6. https://doi.org/10.7176/JIEA/10-3-01.

[20] P. Li, K. Mao, Y. Xu, Q. Li, J. Zhang, Bag-of-Concepts Representation for Document Classification Based on Automatic Knowledge Acquisition from Probabilistic Knowledge Base, Knowl.-Based Syst. 193 (2020), 105436. https://doi.org/10.1016/j.knosys.2019.105436.

[21] J. Choi, S.-W. Lee, Improving FastText with Inverse Document Frequency of Subwords, Pattern Recognit. Lett. 133 (2020), 165–172. https://doi.org/10.1016/j.patrec.2020.03.003.

[22] A. Indrawati, H. Subagyo, A. Sihombing, W. Wagiyah, S. Afandi, Analyzing The Impact of Resampling Method for Imbalanced Data Text in Indonesian Scientific Articles Categorization, BACA: J. Dokument. Inf. 41 (2020), 133-141. https://doi.org/10.14203/j.baca.v41i2.702.

[23] Y. Sun, H. Que, Q. Cai, et al. Borderline SMOTE Algorithm and Feature Selection-Based Network Anomalies Detection Strategy, Energies 15 (2022), 4751. https://doi.org/10.3390/en15134751.

[24] Q. Wang, R.Q. Peng, J.Q. Wang, et al. NEWLSTM: An Optimized Long Short-Term Memory Language Model for Sequence Prediction, IEEE Access 8 (2020), 65395–65401. https://doi.org/10.1109/ACCESS.2020.2985418.

[25] G. Van Houdt, C. Mosquera, G. Nápoles, A Review on the Long Short-Term Memory Model, Artif. Intell. Rev. 53 (2020), 5929–5955. https://doi.org/10.1007/s10462-020-09838-1.

[26] A. Mehboob, M.U. Akram, N.S. Alghamdi, A.A. Salam, A Deep Learning Based Approach for Grading of Diabetic Retinopathy Using Large Fundus Image Dataset, Diagnostics 12 (2022), 3084. https://doi.org/10.3390/diagnostics12123084.

[27] M.H. Rahman, S. Yuan, C. Xie, Z. Sha, Predicting Human Design Decisions with Deep Recurrent Neural Network Combining Static and Dynamic Data, Design Sci. 6 (2020), e15. https://doi.org/10.1017/dsj.2020.12.

[28] R.L. Abduljabbar, H. Dia, P.W. Tsai, Development and Evaluation of Bidirectional LSTM Freeway Traffic Forecasting Models Using Simulation Data, Sci. Rep. 11 (2021), 23899. https://doi.org/10.1038/s41598-021-03282-z.

[29] E. Zvornicanin, Differences Between Bidirectional and Unidirectional LSTM, Baeldung, 2023. https://www.baeldung.com/cs/bidirectional-vs-unidirectional-lstm.

[30] K.S. Nugroho, I. Akbar, A. Suksmawati, et al. Deteksi Depresi Dan Kecemasan Pengguna Twitter Menggunakan Bidirectional LSTM, arXiv:2301.04521 [cs.CL] (2023). https://doi.org/10.48550/ARXIV.2301.04521.

[31] D. Krstinić, M. Braović, L. Šerić, D. Božić-Štulić, Multi-Label Classifier Performance Evaluation with Confusion Matrix, in: Computer Science & Information Technology, AIRCC Publishing Corporation, 2020: pp. 01–14. https://doi.org/10.5121/csit.2020.100801.

[32] A. Burkov, The Hundred - Page Machine Learning Book, Eleven Languages, 2019.

[33] Cinmayii Manliguez, Generalized Confusion Matrix for Multiple Classes, (2016), ResearchGate. https://doi.org/10.13140/RG.2.2.31150.51523.