Journal of Information Systems and Informatics
Vol. 7, No. 2, June 2025 e-ISSN: 2656-4882 p-ISSN: 2656-5935
DOI: 10.51519/journalisi.v7i2.1147 Published By DRPM-UBD

Enhancing Hate Speech Detection: Leveraging Emoji Preprocessing with BI-LSTM Model

Junita Amalia[1], Sarah Rosdiana Tambunan[2], Susi Eva Maria Purba[3], Walker Valentinus Simanjuntak[4]

[1,2,3,4] Faculty of Informatics and Electrical Engineering, Institut Teknologi Del, Indonesia
Email: [1]junita.amalia@del.ac.id, [2]sarah.tambunan@del.ac.id, [3]susi.purba@del.ac.id, [4]iss21012@del.ac.id

Abstract

Microblogging platforms like Twitter enable users to rapidly share opinions, information, and viewpoints. However, the vast volume of daily user-generated content poses challenges in ensuring the platform remains safe and inclusive. One key concern is the prevalence of hate speech, which must be addressed to foster a respectful and open environment. This study explores the effectiveness of the Emoji Description Method (EMJ DESC), which enhances tweet classification by converting emojis into descriptive text or sentences. These descriptions are then encoded into numerical vector matrices that capture the meaning and emotional tone of each emoji. Integrated into a basic text classification model, these vectors help improve detection performance. The research examines how different emoji preprocessing strategies affect the performance of a BI-LSTM model for hate speech classification. Results show that removing emojis significantly reduces accuracy (68%) and weakens the model's ability to distinguish between hate and non-hate speech, due to the loss of valuable semantic context. In contrast, retaining emoji semantics either through textual descriptions or embeddings boosts classification accuracy to 93% and 94%, respectively. The highest performance is achieved through emoji embedding, highlighting its ability to capture subtle non-verbal cues critically for accurate hate speech detection. Overall, the findings emphasize the importance of incorporating emoji-aware preprocessing techniques to enhance the effectiveness of social media content classification.

Keywords: Twitter, Emoji Description, Hate Speech, Emoji Preprocessing, BI-LSTM

1. INTRODUCTION

In the digital communication landscape, social media platforms have become central to the exchange of information, opinions, and emotions. Twitter, a widely used microblogging platform operated by Twitter, Inc., enables users to disseminate brief messages at scale and speed. Expressions in these platforms often extend beyond textual content to include graphic symbols such as emojis, which serve as crucial carriers of emotional nuance, frequently conveying sentiments that are otherwise inexpressible using words alone [1]. While such platforms promote

freedom of expression, they also expose users to the proliferation of hate speech. Hate speech involves the use of language to demean, provoke, or attack individuals or communities based on inherent or ideological characteristics. This can disrupt digital discourse and, in more serious cases, escalate into tangible societal conflict. Advancements in natural language processing (NLP) have led to the development of various techniques for textual classification, particularly for detecting hate speech [2], [3], [4]. However, conventional classification models tend to focus solely on textual elements, often ignoring the semantic contributions of emojis [2]. As a result, emojis are frequently removed or overlooked during preprocessing stages, potentially eliminating valuable affective cues.

To mitigate this, recent research has proposed the incorporation of emojis into classification pipelines. Two notable techniques include Emoji Description (EMJ-DESC), which replaces emojis with corresponding textual descriptions, and Emoji Embedding (EMJ-EMBED), which transforms emojis into vectorized representations [5][6]. These strategies enable models to better interpret emotional content. Nonetheless, the work of Singh et al. [7] was limited to the top 10 most used emojis, thereby failing to capture the full expressive range present in natural datasets.

Other findings suggest that integrating emojis into classification models can yield performance improvements. For instance, a study employing the Naïve Bayes classifier achieved a modest accuracy increase—up to 77.55%—when emojis were retained in the data [8]. Various classification models have been evaluated for hate speech detection. One study using the K-Nearest Neighbors (KNN) algorithm reported an accuracy of 67.86% for identifying hate content on Twitter [6]. In another comparative analysis, the BERT model outperformed traditional classifiers like SVM and logistic regression, particularly when emoji descriptions and sentiment features were added to the textual content. The BERT-based approach achieved F1-scores of 84.3% for offensive language detection, 81.8% for hate speech detection, and 45.1% for fine-grained hate speech categorization (e.g., by race, religion, or social class) [9].

In the domain of deep learning, Long Short-Term Memory (LSTM) and Bidirectional LSTM (Bi-LSTM) architectures have shown considerable success in text classification tasks, including sentiment and hate speech detection [10] [11][12]. A study comparing LSTM and Bi-LSTM models using Word2Vec embeddings demonstrated that Bi-LSTM outperforms LSTM, achieving a peak accuracy of 87% [13]. Additional evaluations corroborated these findings, showing accuracies of 78.67% and 80.25% for LSTM and Bi-LSTM, respectively [8]. The enhanced performance of Bi-LSTM can be attributed to its bidirectional structure, which enables simultaneous forward and backward context processing—enhancing its sensitivity to the sequential dependencies in text [14].

Building on this foundation, the present research explores the impact of emoji preprocessing on hate speech detection by employing Bi-LSTM in three experimental setups: (1) without emoji preprocessing, (2) using emoji descriptions, and (3) using emoji embeddings. In contrast to prior studies that were limited to subsets of commonly used emojis, this study incorporates the complete emoji set available in the dataset. The objective is to enhance the accuracy and robustness of hate speech classification on social media through advanced emoji-aware preprocessing and hyperparameter optimization.

2. METHODS

The research methodology begins with a data preprocessing stage. Data preprocessing is generally known as pre-processing. Data generally, before entering the classification process, consists of raw data that has a lot of noise, is large in size, and comes from various sources. Data Preprocessing is a process or stage that occurs with the aim of managing incoming raw data and converting it into superior data or optimal input to proceed to the next step [14], [15], [16], [17], [18]. Data Cleaning is the step of cleaning data values that are incomplete, correcting data inconsistencies, and minimizing noise when identifying outliers. At this stage, we will discuss how to clean missing data and explain data smoothing techniques [19]. This is followed by three alternative text preprocessing approaches based on emoji handling, which are using emoji embeddings, converting emojis to textual descriptions, or removing emojis entirely. The vectors of the words are then summed to obtain the vector representation of the emoji. This process is carried out for each data pair in the dataset, resulting in a set of emoji vector representations that reflect the emotions associated with those emojis. Emoji embedding will work by generating numerical representations of the emojis used in hate speech [20].

The preprocessed data is then passed to a classification model built with a Bidirectional Long Short-Term Memory (BI-LSTM) architecture. This structure is designed to compare the impact of different emoji processing techniques on hate speech detection performance. For a detailed overview of the process, refer to Figure 1.

2.1. Data Collection

The dataset used in this research comprises English-language tweets sourced from Hugging Face (https://huggingface.co/datasets/HannahRoseKirk). It has 5,912 rows and categorized into two classes, hate speech (50.48%) and non-hate speech (49.52%).

Journal of Information Systems and Informatics
Vol. 7, No. 2, June 2025
p-ISSN: **2656-5935** http://journal-isi.org/index.php/isi e-ISSN: **2656-4882**

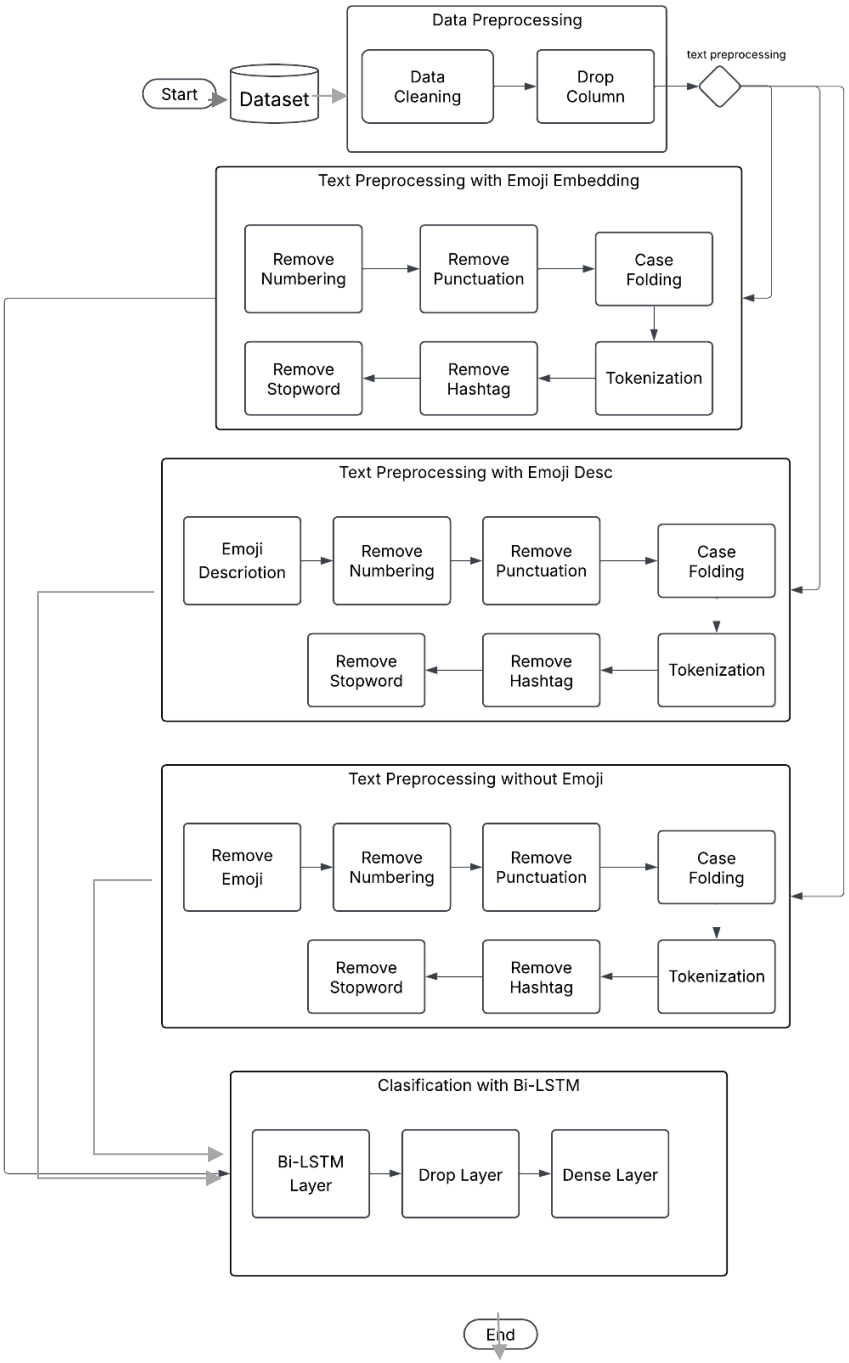

Figure 1. Research Methods

p-ISSN: **2656-5935** http://journal-isi.org/index.php/isi e-ISSN: **2656-4882**

2.2. Data Preprocessing

To prepare the dataset for subsequent analysis, preprocessing steps included data augmentation and column selection. A random swap technique was applied to generate synthetic variations by randomly altering word positions within sentences [21], [22]. This process increased the dataset size to 25,938 rows. In addition, only the relevant columns 'text' and 'label_gold' were retained for analysis, while all other columns were removed. Table 1 illustrates the original tweet and its augmented variants.

Table 1. Augmentation Result

Original Tweet	Augmented Tweet
I would love to 🔪 some 👤	I would love 👤 🔪 some to
	I would love to 👤 some 🔪
	would I love to 🔪 some 👤
	I would 👤 to 🔪 some love
	I to love would 🔪 some 👤

2.3. Text Preprocessing

The text preprocessing stage was conducted to clean and standardize the tweet data prior to analysis and modeling [23], [24]. After completing data preprocessing step, the textual data undergoes further preparation. The process began with loading the dataset, followed by handling emojis based on the chosen approach, either removing them entirely or converting them into descriptive text using emoji description extraction.

In this study we use emoji description; automatically generated based on a hybrid approach combining predefined semantic references and contextual analysis. Initially, each emoji is mapped to its standard meaning using the Unicode Consortium's official short names and annotations (Emoji Desc). These predefined meanings serve as the baseline for interpretation. To enhance relevance and contextual accuracy, we then apply a natural language processing model that adjusts these descriptions based on the surrounding text in which the emoji appears. This allows the final descriptions to reflect both the conventional definition of the emoji and its intended meaning within a specific usage context, ensuring a more nuanced and representative interpretation of emoji semantics in our dataset.

All text was then converted to lowercase to ensure consistency, and punctuation marks were removed to eliminate unnecessary noise. Stopwords were filtered out using the NLTK corpus to retain only meaningful terms, and stemming was

applied using the PorterStemmer to reduce words to their base forms. Hashtags were also removed, as they did not provide significant value for the classification task. Additionally, informal or abbreviated words such as "thx" and "u" were normalized to their formal equivalents like "thanks" and "you." After preprocessing, the dataset was split into training and testing sets with an 80:20 ratio. Finally, the cleaned text was tokenized using the tokenizer() function to convert it into numerical sequences suitable for input into the classification model.

2.4. Text Classification

The classification process employs a Bidirectional Long Short-Term Memory (BI-LSTM) model, which processes input sequences in both forward and backward directions, allowing it to capture contextual dependencies from both past and future words. This bidirectional architecture is particularly effective for interpreting polysemous words and understanding nuanced relationships in text crucial elements in hate speech detection [11], [25]. The Bidirectional Long Short-Term Memory (BI-LSTM) model used in this study consists of two hidden layers, each with 64 LSTM units. The input sequences are first embedded using a pre-trained word embedding layer (GloVe, 300-dimensional), followed by the BI-LSTM layers. A dropout rate of 0.5 is applied after each LSTM layer to prevent overfitting. For the output, a dense layer with a softmax activation function is used to perform classification.

The model is trained using the Adam optimizer with a learning rate of 0.001, and categorical cross-entropy is used as the loss function. We trained the model for 32 over 100 epochs with a batch size of 64, using early stopping with a patience of 3 epochs based on validation loss to prevent overfitting. The model's performance was evaluated using accuracy, precision, recall, and F1-score metrics to ensure a comprehensive assessment of its generalizability across different classes. These parameters were chosen after preliminary tuning and 4-fold cross-validation to balance model complexity with generalization performance.

2.5. Evaluation

After training, the BI-LSTM model was evaluated on both the training and testing datasets using multiple performance metrics, including loss, accuracy, precision, recall, and F1-score. These metrics provide a comprehensive assessment of the model's ability to distinguish between hate speech and non-hate speech. To convert these probabilities into binary class labels, the predicted labels were obtained by selecting the index of the maximum value for each prediction using np.argmax() function. This transformation allows for direct comparison between predicted and actual classes. A classification report was then generated to summarize the model's performance across precision, recall, and F1-score for each

class, providing detailed insights into how well the model generalizes to unseen data.

3. RESULTS AND DISCUSSION

This section provides a comprehensive analysis of the experimental results obtained using the BI-LSTM model for hate speech classification. The study primarily aimed to investigate how various emoji preprocessing strategies affect the classification accuracy and overall model performance. Three distinct approaches were employed: emoji removal, emoji description, and emoji embedding. Each approach was tested using a consistent dataset, and their performances were evaluated based on four widely accepted metrics accuracy, precision, recall, and F1-score. In addition to numerical metrics, illustrative figures and tables were provided to further elucidate the outcomes and facilitate a direct comparison across methods.

3.1. BI-LSTM Without Emoji

The first experimental condition explored the impact of removing emojis entirely from the dataset. This approach aims to assess how the model performs when emojis often strong carriers of sentiment, tone, and context are excluded from the input text. When emojis are deleted, the model is restricted to interpreting only the remaining plain text. As shown in Table 2, which outlines an example before and after emoji deletion, the preprocessing effectively stripped the emojis from the sentences, leaving only the core text structure intact.

Table 2. Text and emoji before and after emoji removal

Before Emoji Deletion	After Emoji Deletion
I would love to 🔪 some 🤮	I would love to some
I would love to 🔪 some 🤬	I would love to some

This minimalist representation results in the loss of valuable semantic and emotional indicators. For example, the emoji 🔪 (knife) can signify aggression or threat, while 👳 (person wearing turban) might imply ethnicity or identity—both of which are crucial for hate speech detection. By removing these elements, the model misses out on understanding the implicit tone and potential hostility embedded in the original message.

The results of this approach, as depicted in Figure 2, were underwhelming. The BI-LSTM model achieved an accuracy of 68%, which is significantly lower compared to the other configurations. Precision and recall were unbalanced, particularly with the model struggling to correctly identify class 0 (non-hate

speech). Although class 1 (hate speech) predictions were marginally better, the overall F1-score remained modest, reflecting a weak equilibrium between false positives and false negatives. This result strongly suggests that emojis contribute essential contextual information that, when omitted, leads to a considerable drop in model performance.

	precision	recall	f1-score	support
0	0.68	0.65	0.67	2680
1	0.67	0.70	0.69	2708
accuracy			0.68	5388
macro avg	0.68	0.68	0.68	5388
weighted avg	0.68	0.68	0.68	5388

Figure 2. Evaluation metrics BI-LSTM without emojis

Moreover, the error distribution across the two classes revealed that the model frequently misclassified non-hateful content as hateful and vice versa, further evidencing its difficulty in discerning subtle contextual cues. These findings highlight a critical shortcoming: when the model is deprived of emotionally charged or sentiment-bearing symbols like emojis, its ability to detect hate speech is significantly impaired.

3.2. BI-LSTM with Emoji Description

The second experiment involved replacing emojis with their descriptive text equivalents. This preprocessing strategy attempts to retain the semantic value of emojis by converting them into phrases that convey their meaning in natural language. For instance, the knife emoji 🔪 is transformed into the phrase "kitchen knife," and 👳 becomes "person wearing turban." This conversion helps in embedding the emoji's connotation directly into the input, making it more digestible for a text-based model.

Table 3 illustrates how this transformation is implemented. The original sentence, when converted, becomes more verbose but now includes contextually rich textual cues that approximate the intent behind the emojis.

Table 3. Text and emoji before and after change to emoji description

Before Emoji Deletion	After Emoji Deletion
I would love to 🔪 some 👳🏽	I would love to kitchen knife some person wearing turban
I would love to 🔪 some 👳🏿	I would love to kitchen knife some man dark skin tone

The outcomes from this strategy, presented in Figure 3, were markedly better than the previous approach. The BI-LSTM model recorded an accuracy of 93%, a significant improvement over the emoji-deletion strategy. In addition, precision, recall, and F1-score all hovered around 92-95%, showcasing strong model confidence and reliability across both hate and non-hate categories.

	precision	recall	f1-score	support
0	0.92	0.95	0.93	2680
1	0.95	0.92	0.93	2708
accuracy			0.93	5388
macro avg	0.93	0.93	0.93	5388
weighted avg	0.93	0.93	0.93	5388

Figure 3. Evaluation metrics BI-LSTM with emoji description

The use of macro-average and weighted-average metrics revealed a balanced performance between the two classes. The model was equally adept at detecting both hate speech and neutral content, suggesting that descriptive phrases successfully preserved the contextual meaning of emojis without introducing noise. Importantly, this method enabled the model to pick up on subtle aggressions or insinuations masked behind emojis particularly relevant in social media posts where hate speech is often veiled or sarcastic.

However, there are trade-offs. While converting emojis into text helps maintain context, it also lengthens the input sequence, which might affect processing time and introduce syntactic ambiguity in certain cases. For example, describing a complex emoji with multiple attributes (e.g., "man dark skin tone wearing sunglasses") can clutter the sentence and potentially obscure its original structure. Still, the results clearly affirm that maintaining the emoji's intent even through verbose means significantly aids the model's interpretive capabilities.

3.3. BI-LSTM with Emoji Embedding

In the third and final experiment, emojis were not deleted or described but were instead converted into embeddings—dense, high-dimensional vectors that capture the semantic and emotional significance of each emoji. This approach allows the BI-LSTM model to understand emojis in the same way it processes words in traditional NLP applications, without requiring any changes to the sentence structure.

Unlike the previous methods, emoji embedding retains the visual symbols and integrates them as learnable components within the model's architecture. This not only preserves the original syntactic form of the sentence but also leverages the

p-ISSN: **2656-5935** http://journal-isi.org/index.php/isi e-ISSN: **2656-4882**

non-verbal cues carried by emojis in a computationally efficient manner. The results of this approach are displayed in Figure 4.

	precision	recall	f1-score	support
0	0.93	0.95	0.94	2680
1	0.95	0.93	0.94	2708
accuracy			0.94	5388
macro avg	0.94	0.94	0.94	5388
weighted avg	0.94	0.94	0.94	5388

Figure 4. Evaluation metrics BI-LSTM with emoji embedding

This method produced the highest overall performance, with the model achieving a remarkable accuracy of 94%. Precision and recall rates were similarly high across both classes, with an F1-score that indicated strong balance between the model's sensitivity and specificity. This suggests that emoji embeddings provide a richer semantic representation, enabling the model to better generalize across diverse and nuanced samples of hate speech.

One significant advantage of this approach is that it maintains the compactness and integrity of the input sentence, unlike the descriptive method that expands and potentially distorts sentence flow. Moreover, because the embeddings are trained in context, they can capture complex relationships such as sarcasm, irony, or threat implication that go beyond literal descriptions.

The results demonstrated that the model not only excelled in classifying overt hate speech but also performed well on ambiguous or borderline cases, a common challenge in real-world applications. The ability to learn from both lexical and non-verbal inputs gives this configuration a distinct edge, particularly for applications that rely on analyzing informal or socially driven content, such as Twitter or Reddit posts.

3.4. Discussion

The comparative analysis of the three preprocessing strategies is visually summarized in Figure 5, which aggregates the performance metrics—accuracy, precision, recall, and F1-score—achieved by the BI-LSTM model under each experimental condition. This synthesis allows us to clearly observe how different treatments of emojis influence the model's ability to detect hate speech within social media texts.

p-ISSN: **2656-5935** http://journal-isi.org/index.php/isi e-ISSN: **2656-4882**

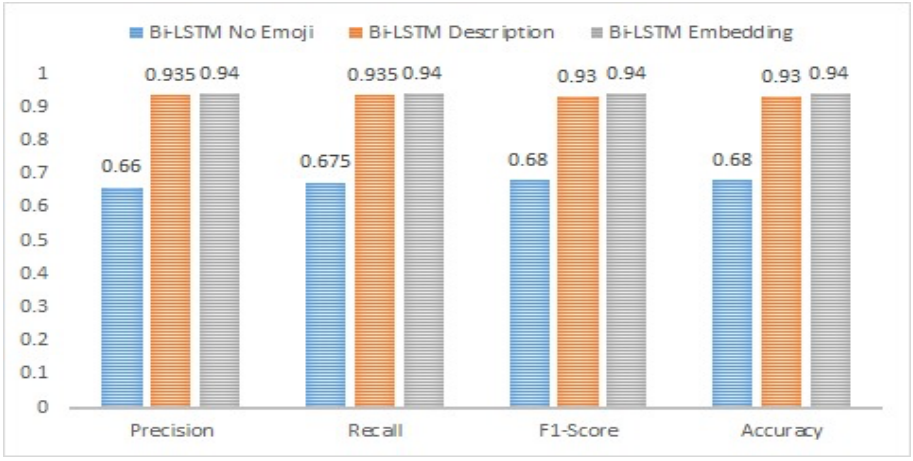

Figure 5. Summarizes the evaluation metrics

From a holistic standpoint, the results underscore the pivotal role that emojis play in shaping digital communication. In contemporary online platforms where messages are often brief, emotionally charged, and semi-structured emojis have evolved into critical contextual and affective indicators. The removal of these elements, as shown in the first experimental setting, significantly degraded the model's performance. With an overall accuracy of just 68%, the model struggled to detect patterns that are often conveyed through subtle or symbolic cues. This suggests that emojis are not mere embellishments or stylistic additions but are in fact semantic components that can shift the tone and interpretability of a message.

By contrast, the second and third experiments emoji description and emoji embedding revealed that preserving emoji semantics dramatically enhances classification effectiveness. When emojis were replaced with their textual descriptions, the model achieved a remarkable 93% accuracy. This approach proved particularly useful for transforming implicit emotional or cultural expressions into explicit linguistic information, allowing the BI-LSTM model to grasp complex nuances that would otherwise be lost. Yet, it comes with certain limitations. One such limitation is the increased verbosity of the input text, which could interfere with the syntactic structure and introduce redundancy. Long-winded phrases like "person wearing turban" or "man with dark skin tone" may affect tokenization and interpretation in sequence-based models, possibly leading to semantic dilution or ambiguity in longer texts.

The most effective strategy, however, was the use of emoji embeddings, which yielded an even higher accuracy of 94%. This method provided the model with dense, semantically meaningful vector representations of emojis without altering the original sentence flow. Unlike textual descriptions, embeddings encapsulate

affective, cultural, and contextual connotations within a fixed-dimensional space, allowing the model to learn nuanced relationships across emoji usage patterns. Importantly, this strategy enhances the model's generalizability and robustness, particularly in detecting veiled or covert forms of hate speech where the textual content may be neutral, but the accompanying emojis drastically alter the meaning.

The implications of these findings are particularly relevant in the domain of automated content moderation and toxicity detection. In many real-world cases, hate speech does not manifest overtly through offensive words but is embedded in subtle linguistic tricks, sarcasm, and emoji usage. Emojis can be weaponized—used sarcastically or in coded sequencesto evade filters or manual review systems. For instance, combining seemingly benign phrases with emojis like 🔪, 🧠, or 💊 can imply violent intent without explicitly stating it. In such contexts, removing or ignoring emojis could render a classifier blind to the latent hostility or aggression embedded in the content.

Furthermore, this discussion reinforces the importance of designing hate speech classifiers that are context-aware and multimodal in nature. Emojis bridge the gap between text and emotion, acting as carriers of intent that are easily overlooked in conventional NLP systems. Incorporating emoji embeddings enables a more holistic understanding of user intent, particularly when coupled with models like BI-LSTM, which are designed to capture sequential dependencies and temporal patterns. From an architectural perspective, emoji embeddings also offer scalability. Once integrated into the model's vocabulary, they function just like any other word embedding, making them ideal for large-scale deployment without necessitating additional preprocessing pipelines. This feature is particularly valuable for social platforms that require real-time monitoring of vast streams of user-generated content.

The results clearly validate that emoji-aware processing is not merely an enhancement but a necessity for achieving high-performance hate speech detection. The nuanced affective and symbolic layers provided by emojis significantly influence model predictions. Among the three strategies tested, emoji embedding emerged as the most efficient and scalable approach, offering a superior balance between semantic fidelity and structural integrity. Future work in this area could explore combining emoji embeddings with multimodal features like image metadata, sentiment scores, or user profiling to further improve detection accuracy in real-world scenarios.

4. CONCLUSION

This study demonstrates that the choice of emoji preprocessing strategy has a significant impact on the performance of the BI-LSTM model in hate speech

classification tasks. Specifically, the results reveal that removing emojis entirely leads to the poorest performance, highlighting the loss of critical contextual and emotional cues essential for accurately interpreting user intent. In contrast, methods that retain and represent emoji semantics either by converting them into textual descriptions or incorporating them as embeddings markedly enhance classification accuracy. Among the three approaches evaluated, emoji embedding proved to be the most effective, offering the highest accuracy and the most balanced performance across evaluation metrics. This suggests that preserving emoji-related information is crucial for building reliable and context-aware hate speech detection models, particularly in the domain of social media where emojis often serve as proxies for tone, sarcasm, and sentiment. Moreover, while the findings are promising, it's important to acknowledge the potential sensitivity of the model to emoji distribution and class imbalance, factors not deeply explored in this single-dataset study. Future research should examine the model's generalizability by applying it to diverse datasets with varying emoji usage patterns and hate speech frequencies. Such studies would provide deeper insights into the robustness of emoji-aware models across different cultural, linguistic, and platform-specific contexts, thereby enhancing the practical application of hate speech detection systems in real-world settings.

REFERENCES

[1] V. B. Lestari, E. Utami, and Hanafi, "Combining Bi-LSTM and Word2vec Embedding for Sentiment Analysis Models of Application User Reviews," *Indonesian Journal of Computer Science*, vol. 13, no. 1, pp. 312–326, 2024, doi: 10.33022/ijcs.v13i1.3647.

[2] A. Salau and T. K. Yesufu, "Recent Trends in Image and Signal Processing in Computer Vision," unpublished, Dec. 2020.

[3] Y. A. Jasim, M. G. Saeed, and M. B. Raewf, "Analyzing Social Media Sentiment: Twitter as a Case Study," *Advances in Distributed Computing and Artificial Intelligence Journal*, vol. 11, no. 4, pp. 427–450, 2022, doi: 10.14201/adcaij.28394.

[4] M. A. Fauzi and A. Yuniarti, "Ensemble method for Indonesian Twitter hate speech detection," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 11, no. 1, pp. 294–299, 2018, doi: 10.11591/ijeecs.v11.i1.pp294-299.

[5] S. W. Azumah, N. Elsayed, Z. ElSayed, M. Ozer, and A. La Guardia, "Deep Learning Approaches for Detecting Adversarial Cyberbullying and Hate Speech in Social Networks," arXiv preprint, 2024. [Online]. Available: http://arxiv.org/abs/2406.17793

[6] O. Adel, K. M. Fathalla, and A. Abo ElFarag, "MM-EMOR: Multi-Modal Emotion Recognition of Social Media Using Concatenated Deep Learning Networks," *Big Data and Cognitive Computing*, vol. 7, no. 4, 2023, doi: 10.3390/bdcc7040164.

[7] A. A. Arifiyanti and E. D. Wahyuni, "Emoji and emoticon in tweet sentiment classification," in *Proc. 6th Information Technology International Seminar (ITIS)*, 2020, pp. 145–150, doi: 10.1109/ITIS50118.2020.9320988.

[8] M. Amrullah, I. Budi, A. Santoso, and P. Putra, "The effect of using Emoji and Hashtag in sentiment analysis on Twitter case study: Indonesian online travel agent," in *AIP Conference Proceedings*, vol. 2023, p. 20013, 2023, doi: 10.1063/5.0118228.

[9] M. J. Althobaiti, "BERT-based Approach to Arabic Hate Speech and Offensive Language Detection in Twitter: Exploiting Emojis and Sentiment Analysis," *International Journal of Advanced Computer Science and Applications*, vol. 13, no. 5, pp. 972–980, 2022, doi: 10.14569/IJACSA.2022.01305109.

[10] U. Ite, "Perbandingan IndoBERT dan Bi-LSTM Dalam Mendeteksi Pelanggaran," *Jurnal Teknologi dan Sistem Komputer*, vol. 8, no. 1, pp. 52–59, 2025.

[11] E. Aurora, A. Zahra, Y. Sibaroni, and S. Prasetyowati, "Classification of Multi-Label of Hate Speech on Twitter Indonesia using LSTM and BiLSTM Method," *JINAV: Journal of Information and Visualization*, vol. 4, no. 2, pp. 2746–1440, 2023, doi: 10.35877/454RI.jinav1864.

[12] B. Jang, M. Kim, G. Harerimana, S. U. Kang, and J. W. Kim, "Bi-LSTM model to increase accuracy in text classification: Combining word2vec CNN and attention mechanism," *Applied Sciences*, vol. 10, no. 17, 2020, doi: 10.3390/app10175841.

[13] A. R. Gunawan, R. Faticha, and A. Aziza, "Sentiment Analysis Using LSTM Algorithm Regarding Grab Application Services in Indonesia," *Jurnal Teknologi dan Sistem Komputer*, vol. 9, no. 2, pp. 322–332, 2025.

[14] V. Prasetyo and A. Samudra, "Hate speech content detection system on Twitter using K-nearest neighbor method," in *AIP Conference Proceedings*, vol. 2022, p. 50001, 2022, doi: 10.1063/5.0080185.

[15] K. Keykhosravi, A. Hamednia, H. Rastegarfar, and E. Agrell, "Data preprocessing for machine-learning-based adaptive data center transmission," *ICT Express*, vol. 8, no. 1, pp. 37–43, 2022, doi: 10.1016/j.icte.2022.02.002.

[16] K. Maharana, S. Mondal, and B. Nemade, "A review: Data pre-processing and data augmentation techniques," *Global Transitions Proceedings*, vol. 3, no. 1, pp. 91–99, 2022, doi: 10.1016/j.gltp.2022.04.020.

[17] N. Pandey, P. K. Patnaik, and S. Gupta, "Data Pre Processing for Machine Learning Models using Python Libraries," *International Journal of Engineering and Advanced Technology*, vol. 9, no. 4, pp. 1995–1999, 2020, doi: 10.35940/ijeat.d9057.049420.

[18] P. Gong, Y. Ma, C. Li, X. Ma, and S. H. Noh, "Understand Data Preprocessing for Effective End-to-End Training of Deep Neural Networks," arXiv preprint, 2023. [Online]. Available: http://arxiv.org/abs/2304.08925

[19] J. Han, M. Kamber, and J. Pei, *Data Mining: Concepts and Techniques*, 3rd ed. Elsevier, 2011, doi: 10.1016/C2009-0-61819-5.

[20] L. Saragih, M. Nababan, Y. Simatupang, and J. Amalia, "Analisis Self-Attention Pada Bi-Directional LSTM Dengan Fasttext Dalam Mendeteksi Emosi Berdasarkan Text," *Zo. Jurnal Sistem Informasi*, vol. 4, no. 2, pp. 144–156, 2022, doi: 10.31849/zn.v4i2.10846.

[21] L. F. A. O. Pellicer, T. M. Ferreira, and A. H. R. Costa, "Data augmentation techniques in natural language processing," *Applied Soft Computing*, vol. 132, p. 109803, 2023, doi: 10.1016/j.asoc.2022.109803.

[22] D. Wang and J. Eisner, "Synthetic data made to order: The case of parsing," in *Proc. 2018 Conf. Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1325–1337, 2018, doi: 10.18653/v1/d18-1163.

[23] D. Raka, V. Saputra, and E. R. Arumi, "Optimizing Aspect-Based Sentiment Analysis for Kyai Langgeng Park Using PSO and SVM," *Jurnal Ilmu Sistem Informasi*, vol. 6, no. 4, pp. 2856–2867, 2024, doi: 10.51519/journalisi.v6i4.930.

[24] A. Novanto and D. Indra, "Analisis Pre-processing Sentimen Terhadap Komentar Layanan Indihome pada Twitter," *Jurnal Teknologi dan Sistem Informasi*, vol. 5, no. 1, pp. 30–36, 2024.

[25] A. P. J. Dwitama, D. H. Fudholi, and S. Hidayat, "Indonesian Hate Speech Detection Using Bidirectional Long Short-Term Memory (Bi-LSTM)," *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 7, no. 2, pp. 302–309, 2023, doi: 10.29207/resti.v7i2.4642.