ENTHUSIASTIC
INTERNATIONAL JOURNAL OF APPLIED STATISTICS AND DATA SCIENCE
Volume 6, Issue 1, April 2026, pp. 82-94

Identification of Sexual Harassment Comment on Tiktok Platform Using Indobert Embedding and Long Short-Term Memory

Najla Lathifah Mumtaz [a,1], Ayundyah Kesumawati [a,2,*], Arum Handini Primandari [a,b,3]

[a] Statistics Department, Universitas Islam Indonesia, Jl. Kalirang 14.5, Sleman, Yogyakarta 55584, Indonesia
[b] School of Mathematics and Statistics, Victoria University of Wellington, Cotton Building, Kelburn, Wellington 6140, New Zealand
[1] 21611041@students.uii.ac.id; [2] ayundyah.k@uii.ac.id*; [3] arumhandini.primandari@vuw.ac.nz
* Corresponding author

ARTICLE INFO

History
Submitted: February 16, 2026
Revised: May 1, 2026
Accepted: May 7, 2026

Keywords
Text Classification
IndoBERT Embedding
Long Short-Term Memory
Sexual Harassment Comment
SMOTE

ABSTRACT

Sexual harassment in online comment sections is a growing concern on social media platforms like TikTok, where informal language makes manual moderation ineffective. This study developed an automated detection model using a hybrid Indonesian bidirectional encoder representation from transformers (IndoBERT)-long short-term memory (LSTM) architecture, employing IndoBERT as a static feature extractor and an LSTM network to model sequential dependencies. Given the high-class imbalance in social media data, this study specifically evaluated the impact of the synthetic minority over-sampling technique (SMOTE) on classification performance. Experimental results showed that the base IndoBERT-LSTM model achieved a high overall accuracy of 89.04% but struggles with a low recall (0.40) for the minority harassment class. While applying SMOTE improved the model's sensitivity (recall) for harassment to 0.54, it resulted in a significant decrease in precision, and an overall accuracy drop to 87.79%. These findings indicate that while oversampling can modestly enhance the detection of harassment instances, it introduces a substantial trade-off by increasing false positives. This study concludes that for highly informal and imbalanced TikTok data, standard oversampling techniques such as SMOTE may be less effective, suggesting the need for more advanced contextual augmentation or cost-sensitive learning approaches in future digital safety research.

1. Introduction

The advancement of information technology has brought significant changes to human life, especially in the way people communicate and interact with one another. The presence of technology in communication has made it easier to send, deliver, and receive messages through various tools and media, such as the internet and social media. According to data from We Are Social, Indonesia ranks 4th in the world with 167 million active social media users, covering more than 68% of the population [1], [2]. While social media facilitates communication, its irresponsible use can lead to the spread of unverified information, cyberbullying, and various forms of online harassment, all of which can have serious impacts on users' mental health and social well-being [3], [4].

In Indonesia, from January 1, 2024, until now, there have been 20,701 recorded cases of sexual violence, with more than 30% involving sexual harassment, and 79% of the victims being women [5]. Based on the survey results, 756 senior high school students in Indonesia reported experiencing online harassment (*kekerasan berbasis gender online*) in 2021, highlighting the substantial prevalence of online gender-based violence among students [6]. Sexual harassment can be categorized into two types: verbal and nonverbal. Nonverbal harassment typically involves physical acts such as touching or groping that make the victim feel uncomfortable, while verbal harassment includes statements or words that pressure or humiliate the victim. One form of verbal sexual harassment on social media is comments with sexual content directed at someone's body or appearance, sexually suggestive teasing, and aggressive sexual remarks, whether explicit or implicit [7]. One of the most popular social media platforms today is TikTok, which allows users to share short videos that can be widely viewed by others.

The essential challenge in automated text analysis is the transformation of raw text into a numerical representation that preserves the intricate relationship between words and their context. In this study, the Indonesian bidirectional encoder representations from transformers (IndoBERT) model was utilized due to its proven capability in capturing nuanced semantic and contextual meanings within the Indonesian language [8]. Unlike traditional word embeddings, IndoBERT leverages a transformer-based architecture that generates deeply contextualized vectors, ensuring that each word's representation is informed by its surrounding text [9]. By effectively mapping these linguistic nuances into a high-dimensional semantic space, IndoBERT provides a robust foundation for the long short-term memory (LSTM) network to learn and distinguish complex patterns of sexual harassment, which often rely on subtle contextual cues rather than isolated keywords.

Subsequently, the LSTM algorithm was employed as the classification model. LSTM is a type of recurrent neural network (RNN) designed to address short- and long-term memory issues in sequential data and is capable of recognizing important patterns and context in text. The strength of LSTM in text analysis lies in its ability to overcome the vanishing gradient problem and retain relevant information over both short and long periods. Recent Indonesian natural language processing (NLP) work increasingly combines IndoBERT embeddings with LSTM-family architectures as classifiers or sequence encoders for text classification tasks. The core idea is to use IndoBERT as a powerful contextual feature extractor, then let LSTM model longer-range sequential patterns and provide a lighter-weight or more flexible classification head.

A foundational study on Indonesian fake-news classification replaced the traditional embedding layer with IndoBERT, feeding its contextualized token representations into an LSTM classifier. This transfer-learning approach substantially outperformed convolutional neural network (CNN)+IndoBERT and standalone CNN/LSTM, reaching about 97.8% accuracy and showing that LSTM captured IndoBERT's representations more effectively than CNN for this task [10], [11]. The use of BERT embedding with two-layer LSTM also succeed in classifying English language sentiment analysis on social media [12].

This study employed IndoBERT to vectorize textual data and an LSTM-based model to classify comments as containing sexual harassment or not. The data were collected from TikTok, selected due to its rapidly increasing popularity in recent years.

2. The Research Method

2.1. Data Description

The data used in this study were obtained by scraping comments from the TikTok platform, specifically from videos featuring women dancing. A total of 32,939 comments were collected from 61 such TikTok videos. The data collection process was conducted using the Instant Data Scraper browser extension to extract comments directly from the platform.

Data collection for this study was conducted in strict adherence to ethical research standards for social media analytics. First, all data gathered from TikTok consisted of publicly available

p-ISSN 2798-253X
e-ISSN 2798-3153

comments; no private messages or restricted-access content were collected, ensuring compliance with TikTok's terms of service regarding public data usage.

Second, to protect users' privacy and anonymity, a rigorous de-identification process was applied. All personally identifiable information (PII), including usernames (@mentions) and profile links, was removed during preprocessing. Only the raw text of the comments was utilized for model training and analysis. Furthermore, given the sensitive nature of sexual harassment content, the dataset was stored on a secure storage that accessible only to the research team.

2.2. Research Method

This study followed a systematic workflow to detect sexually harassing comments on TikTok. The process began with data collection and manual labeling following by text preprocessing which included case folding, noise removal, stopword removal, and lemmatization. After preprocessing, a vectorization process was required to convert words and sentences into numerical representations. In this study, the vectorization method used was the IndoBERT embedding, which captures the semantic meaning and context of words. Research flowchart is shown in Fig. 1.

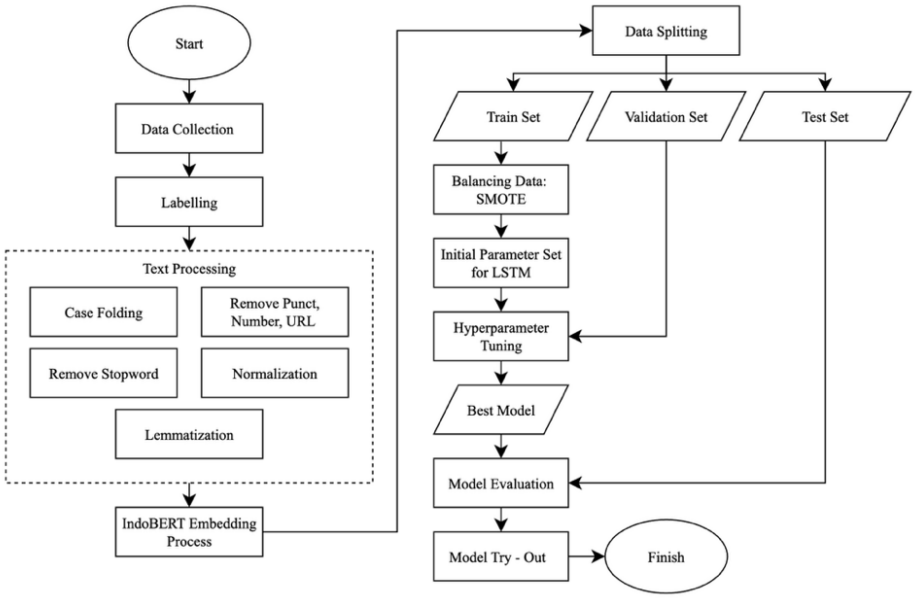


Fig. 1 Research flowchart.

The proposed model employed a hybrid architecture operating in two distinct stages.

a. Feature extraction stage: The cleaned, normalized text was tokenized and fed into a pretrained IndoBERT model. IndoBERT was utilized as a static feature extractor. The model weights were frozen, and the final layer hidden states were extracted to produce 768-dimensional static embedding vectors representing the semantic context of each comment.
b. Sequence modeling stage: These static embeddings were then fed into an LSTM network. The LSTM was trained to learn the sequential dependencies of the comment structure, using the precomputed IndoBERT vectors as fixed inputs. By separating the feature extraction from the sequence modeling, the linguistic contextual power of IndoBERT was combined with the temporal dependency handling of the LSTM.

To address the class imbalance inherent in social media harassment data, the synthetic minority over-sampling technique (SMOTE) was employed. Crucially, SMOTE was applied only to the static embedding vectors generated by IndoBERT for the training dataset. This process was conducted after feature extraction but before sequence modeling, allowing the model to learn from a balanced representation of classes. To prevent data leakage and ensure a realistic evaluation, SMOTE was not applied to the validation or test sets; these sets remained in their original, "actual" vector form to represent real-world data distributions.

3. Method

The methods used included the embedding technique, the balancing technique, and the classification model. Moreover, the metric evaluation employed were accuracy, F1-score, precision, and recall.

3.1. Text Preprocessing

Text preprocessing is an essential step in preparing raw textual data for analysis. Raw text is typically unstructured and contains various types of noise, such as punctuation, numbers, special characters, and informal word forms. Therefore, preprocessing is performed to clean and standardize the text so that only meaningful and relevant information remains, making subsequent analysis more effective.

In the first stage, case folding converted all text to lowercase, ensuring uniformity across tokens. This stage was followed by the removal of noise, including punctuation, numerical characters, and URLs, which did not contribute meaningful semantic information to the classification task. Subsequently, stopword removal was performed to eliminate frequently occurring but semantically insignificant words (e.g., conjunctions and common particles). The text was then normalized, with informal expressions, slang, abbreviations, and misspellings that commonly found on social media, converted to their standard Indonesian forms. This step is particularly important given the informal and highly variable linguistic style of TikTok comments. Finally, lemmatization reduced words to their base or root forms, allowing the model to treat different morphological variants of a word as a single representation. This step helps reduce dimensionality and enhances semantic consistency [13], [14].

3.2. IndoBERT Embedding

Word embedding is a method used to transform words into numerical representations in the form of vectors. This technique allows words with similar contexts to have related vector representations. Since machine learning models can only work with data in vector form, this transformation is essential. Word embedding serves as an effective unsupervised learning application, as it does not require complex data labeling. One approach to word embedding is IndoBERT. BERT is one of the most advanced models for understanding word context and representation. According to [15] BERT has evolved to support multiple languages, including Indonesian, with a specialized version known as IndoBERT. IndoBERT demonstrates superior performance in capturing the nuances of the Indonesian language compared to the general BERT model [16]. The stages of applying IndoBERT are exhibited in in Fig. 2 [17].

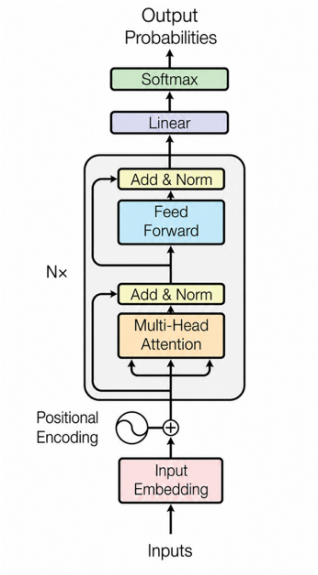


Fig. 2 BERT encoder embedding.

In the input stage, special tokens are added, such as [CLS] at the beginning to represent the entire sentence, [SEP] to separate or end a sentence, and [PAD] to standardize the sequence length. The sentence is then tokenized based on the model's vocabulary, and any unknown words are broken down into sub-tokens marked with a #. The result of tokenization is converted into Token IDs (numerical representations of tokens) and an attention mask (a binary indicator where 1 marks valid tokens and 0 marks padding). The attention mask helps the model focus only on the meaningful tokens and ignore the padding.

BERT consists of 12 layers, each utilizing a self-attention mechanism, which allows the model to understand the context of each word within a sentence. Each word is represented by three key vectors: query (Q), key (K), and value (V)—produced by multiplying the input with trained weight matrices. This mechanism enables the model to focus on the most relevant words in the sentence's context. The output of the attention mechanism in BERT's architecture is a tensor with specific dimensions that reflects how each word has been processed through self-attention. This self-attention output can then be used for various downstream tasks in the processing pipeline, such as text classification, feature extraction, or as input for subsequent layers in more complex model architectures.

3.3. Synthetic Minority Over-Sampling Technique

SMOTE is a method used to address class imbalance by generating synthetic data for the minority class. This technique was introduced in previous study [18] and works by employing a k-nearest neighbor (KNN) approach to generate new samples based on the nearest neighbors of the minority class data [19]. The value of k is determined based on practical considerations or ease of implementation. The SMOTE technique is mathematically defined by (1) [20].

$$X_{syn} = X_i + (X_{KNN} + X_i) \times \delta. \quad (1)$$

SMOTE generates synthetic data by randomly selecting a sample from the minority class X_i, then identifying its nearest neighbor X_{KNN} from the same class. The difference between X_i and X_{KNN} is multiplied by a random ratio between 0 and 1, and the result is added to X_i to form a new data point X_{syn}. This technique creates new samples between existing minority data points, resulting in a more balanced class distribution.

3.4. Long Short-Term Memory

LSTM is a special type of RNN designed to capture and learn long-term dependencies. This concept was first introduced by Hochreiter and Schmidhuber in 1997 and has since been developed and popularized through numerous follow-up studies by various researchers. LSTM has proven to be highly effective in handling a wide range of problems and is now widely used in various applications.

LSTM is specifically designed to address the challenge of managing long-term dependencies, with a built-in ability to retain information over extended periods. This feature makes LSTM particularly effective for tasks that require long-term memory. While its structure is similar to that of a traditional RNN, LSTM incorporates a more complex repeating module. Instead of having a single neural network layer, LSTM uses four interacting layers that are carefully coordinated in their operation [21].

Forget gate determines whether certain information in the cell state should be retained or discarded. The forget gate equation is presented in (2) [21]–[23].

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f. \quad (2)$$

Input gate is responsible for adding new information to the cell state, using a combination of the sigmoid and tanh activation functions. The input gate equation is defined in (3).

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i). \quad (3)$$

After the input gate, a candidate vector for new values to be added to the cell state is created using the tanh function, as shown in (4).

$$\widetilde{C_t} = tanh\,(W_c \cdot [h_{t-1}, x_t] + b_c). \quad (4)$$

Cell state combines the previous state with the output from the forget and input gates.

$$C_t = f_t \cdot C_{t-1} + i_t \cdot \widetilde{C_t}. \quad (5)$$

The output gate generates the output that will be passed to the next hidden layer. It is calculated using (6).

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o. \quad (6)$$

Then, the final output of the LSTM cell is defined in (7).

$$h_t = o_t \times tanh(C_t). \quad (7)$$

3.5. Evaluation Metric

A confusion matrix is a tool used to evaluate the performance of a classification model based on test data for which the true labels are already known. It consists of four main components arranged in a simple table format, allowing us to visualize and assess the model's prediction results. To evaluate the performance of a model, several metrics can be calculated, including accuracy, precision, recall, and F1-score. Accuracy is the metric measures how accurately the model classifies data correctly. It is calculated using the following formula [22].

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN}. \quad (8)$$

where *TP* represents true positive (predicted to be positive and the actual value is also positive), *TN* represents true negative (predicted to be negative and the actual value is also negative), *FP* represents false positive (predicted to be positive but the actual value is negative), and *FN* represents false negative (predicted to be negative but the actual value is positive). Precision measures how accurate the model is in predicting positive classes, by comparing the number of correct positive predictions to the total number of positive predictions made.

$$Precision = \frac{TP}{TP+FP}. \quad (9)$$

Recall measures how well the model is able to detect all actual positive instances.

$$Recall = \frac{TP}{TP+FN}. \quad (10)$$

F1-score evaluates the balance between precision and recall. It is especially useful when dealing with imbalanced datasets.

$$F1 - score = 2 \times \frac{recall \times precision}{recall + precision}. \quad (11)$$

4. Results and Discussion

The result and discussion start with the explanation of obtaining data using scrapping technique followed by the discussion of model result.

4.1. Data Srapping and Preprocessing

The data used in this study were obtained from comments on the TikTok platform, specifically from video content featuring women dancing. A total of 32,939 comments were collected through a scraping process. Based on the results of data scraping, five variables were identified in and presented in Table 1.

Table 1. Results of Comment Data Scraping from the TikTok Platform

User Name	Comment	Date	Like	Reply
fahmi saputra	*emg gua cowo apa kabar Alhamdulillah sehat* (I actually a man what's up (praise) healthy)	7/27/2023	622	View 5 replies
reybukancleng	*jumat berkah #sabar lipeli* (Holly Friday #patient)	7/28/2023	10	View 2 replies
....	...	...	...	...
HAMBA ALLAH	*500net kak plus nasi gorengðŸ˜,* (Five hundred with fried rice)	12/16/2024	18	View 11 replies
Axel	*kasian cowonya* (Poor guy)	12/16/2024	8	View 6 more

The preprocessing aimed to clean the collected text data by removing elements that were not relevant for analysis. The preprocessing process plays a crucial role in the overall performance of the model in later stages, as clean and structured data enables the model to learn patterns more accurately. Table 2 shows steps involved in the text preprocessing process.

Table 2. Example of Data Preprocessing

Comment	Case Folding	Cleaning Data	Tokenizing	Normalization	Remove Stopword	Stemming
cantiknya WANITA PENGHIBUR	*cantiknya wanita penghibur*	*cantiknya wanita penghibur*	*['cantiknya', 'wanita', 'penghibur']*	*['cantiknya', 'wanita', 'penghibur']*	*['cantiknya', 'wanita', 'penghibur']*	*cantik wanita hibur*
greget bngttt samaa mukanyaaðŸ˜	*greget bngttt samaa mukanyaaðŸ˜*	*greget bngt sama mukanya*	*['greget', 'bngt', 'samaa', 'mukanya']*	*['greget', 'banget', 'sama', 'wajahnya']*	*['greget', 'banget', 'sama', 'wajahnya']*	*greget banget sama wajah*
....	....	....	....	....	....	....
jaualan sambil promosi goyang yaa biar bnyak yg beliðŸ— ¿ðŸ™□	*jaualan sambil promosi goyang yaa biar bnyak yg beliðŸ— ¿ðŸ™□*	*jaualan sambil promosi goyang ya biar bnyak yg beli*	*['jaualan', 'sambil', 'promosi', 'goyang', 'ya', 'biar', 'bnyak', 'yg', 'beli']*	*['jualan', 'sambil', 'promosi', 'goyang', 'ya', 'biar', 'banyak', 'yang', 'beli']*	*['jualan', 'sambil', 'promosi', 'goyang', 'biar', 'banyak', 'beli']*	*jual sambil promosi goyang biar banyak beli*

After the stopword removal process, the number of comments was reduced from 28,826 to 23,452.

4.2. Data Label

After the preprocessing stage, the data were manually labeled into two categories: sexual harassment (label 1) and nonsexual harassment (label 0). The labeling process was performed by the researcher as an independent annotator and native speaker of Indonesian language, guided by predefined indicators of sexual harassment, including degrading expressions, comments on physical appearance, explicit sexual language, and inappropriate forms of address. In addition, newly emerging terms commonly used in harassing contexts—such as *"tobrut," "pulen," "plat Jambi," and "tesla logo"*—were also considered during the annotation process to ensure contextual relevance and labeling accuracy (Table 3).

Table 3. Example of Data Labelling

Comments	Label
cantik wanita hibur (beautiful entertaining woman)	1
greget banget sama wajah (really excited about her face)	0
jual sambil promosi goyang biar banyak beli (selling while promoting her body and dancing to get lots of buyers)	1
bisa mantap begitu pinggul (hip can be so nice)	1

p-ISSN 2798-253X
e-ISSN 2798-3153

The results of the manual labeling process showed that 20,087 comments were labeled as nonsexual harassment, while 3,365 comments were labeled as sexual harassment.

4.3. Visualization Using Word Cloud

After the labeling process, the data were visualized using word clouds to identify frequently occurring words in each category. Words that appear more often are displayed in larger sizes, allowing for a visual analysis of language patterns in both sexual harassment and non-harassment comments. The word cloud visualizations for both classes are presented in Fig. 3.

Fig. 3 Word cloud of comments in both classes, (a) sexual harassment, (b) nonsexual harassment.

Based on Fig. 3, there is a striking difference in word usage between sexual harassment and non-harassment comments. Comments labeled as 1 were dominated by sexually nuanced words such as *"hyper," "geol," and "tobrut,"* while comments labeled as 0 more commonly contained polite words like "*cantik"* and *"kamu"* which are appreciative in nature and lack demeaning intent.

4.4. Data Embedding

This study used the IndoBERT-base-p1 model from Hugging Face, which is a BERT variant specifically trained on Indonesian language data. The embedding process involved token, segment, and position embeddings, each with a dimension of 768. Sentences were tokenized with the addition of special tokens: [CLS], [SEP], and [PAD], and the maximum sequence length was set to 63 to optimize memory usage. The embedding output from IndoBERT's last hidden state was represented as a vector of shape (number of documents, 768), which was then averaged using mean pooling to generate a sentence vector. This approach allows the model to effectively capture contextual meaning, even for the same word used in different contexts. The example of IndoBERT is as shown in Table 4.

Table 4. Examples of IndoBERT Embedding Result

Sentence	Embedding Result
sehat sehat perempuan suka goyang perempuan hibur mata laki laki	tensor([-2.1518e-01, 1.4314e+00, 3.2263e-01, -3.1300e-01, 2.1172e-01, -1.5809e-01, -1.2062e+00, -2.8033e-01, 3.9544e-01, 1.0164e-01, -6.0565e-01, -1.1250e-02, -7.9885e-01, 7.2336e-01, -1.5686e-01, ... -8.1115e-01, -7.3677e-01, -3.3797e-01, 2.0710e-01, 2.3932e-01, -7.3027e-01, 4.9618e-01, 4.5150e-01, -5.3525e-02, 1.2225e+00, 1.2417e+00, -6.5356e-01, -9.3395e-01])

4.5. Splitting Data

After the embedding process, the dataset was divided into three subsets: training, validation, and testing. The independent variables consist of the embedded comment vectors, while the target variable represents the corresponding class labels. Data partitioning was performed in two stages. First, the dataset was split into 80% training data and 20% testing data using the train_test_split

function with random_state = 42 to ensure reproducibility. Subsequently, 20% of the training data was allocated as validation data during model training using validation_split = 0.2 as shown in Table 5.

Table 5. Amount of Data for Every Split

Data Type	Total of Data
Data train	15,009 data
Data test	4,860 data
Data validation	3,752 data

The final data split resulted in 64% training data, consisting of 15,009 samples, 20% test data consisting of 4,860 samples, and 16% validation data consisting of 3,752 samples.

4.6. SMOTE Implementation

The class distribution in the training data showed an imbalance, with 85.7% of the comments labeled as nonsexual harassment and only 14.3% labeled as sexual harassment. Since the sexual harassment class is the primary focus of this study, the SMOTE method was applied to address the imbalance by generating synthetic data, resulting in a more balanced class distribution. The changes in data distribution after the balancing process are shown in Table 6, which highlights an increase in the number of comments in the sexual harassment class.

Table 6. Frequency Comparison After SMOTE

Class	Before SMOTE	After SMOTE
Sexual harassment	16,069 data	16,069 data
Nonsexual harassment	2,692 data	16,069 data

4.7. LSTM Model

In the LSTM model development stage, parameters such as the number of LSTM units, dropout rate, dense layer size, and learning rate were tested through several scenarios to obtain the optimal model performance based on evaluation metrics. The parameter values tested can be found in Table 7.

Table 7. LSTM Unit Parameters

Parameter	Unit
Unit LSTM	512 and 256, 256 and 128, 128 and 64
Dropout rate	0.2 ; 0.5 ; 0.7
Learning rate	0.001; 0.0001; 0.002; 0.0002; 0.005; 0.0005
Dense layer	128, 64, 32
Activation Layer in the dense Layer	ReLu and Sigmoid

The LSTM model was designed to classify comments as either sexual harassment or nonsexual harassment. It utilized the embedding output from IndoBERT as input, with a dimension of 1 × 768. The first layer was an LSTM layer with return_sequences=True, allowing sequential outputs to be passed to the next layer. The model included two dropout layers to prevent overfitting, an additional LSTM layer, and two dense layers. The ReLU activation function was used in the first dense layer to capture nonlinear patterns, while a sigmoid function was applied in the second dense layer to output binary probabilities. The model was compiled using binary cross-entropy as the loss function, the Adam optimizer, and accuracy as the evaluation metric. To obtain the best parameter combination, hyperparameter optimization was performed using the GridSearchCV method based on Hyperband, with the search space defined using hp.Choice(). The tuning process was carried out for a maximum of 15 epochs with a factor of 3, producing 30 optimal hyperparameter combinations, detailed in Table 8. The experiment compared model performance using two training datasets, with

and without SMOTE. From 30 hyperparameter combinations, the best configurations were selected based on validation accuracy.

Table 8. Comparison of Model Parameters With and Without SMOTE

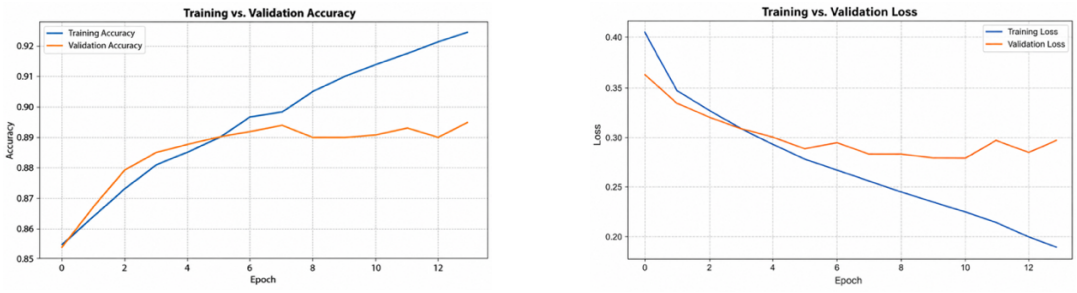
	Units1	Units2	Dropout Rate	Dense Units	Learning Rate	Tuner/ Epochs	Validation Accuracy
Without SMOTE	128	128	0.5	32	0.0002	15	0.900346
SMOTE	256	256	0.5	32	0.0002	15	0.995192

Next, the model was retrained using the training data with the .fit() function for a maximum of 15 epochs. To prevent overfitting, early stopping was applied, causing training to stop automatically if no improvement occurred for three consecutive epochs, and the best weights were retained. A batch size of 32 was used, as it is efficient for the training process. Table 9 presents the results of the LSTM model training and its evaluation on the data validation.

Table 9. Comparison of Model Performance With and Without SMOTE

	Accuracy	Loss	Validation Accuracy	Validation Loss	Epoch	Timestep
Without SMOTE	0.9274	0.1891	0.8953	0.2964	14	15ms/step
SMOTE	0.9735	0.0760	0.9813	0.0608	9	50ms/step

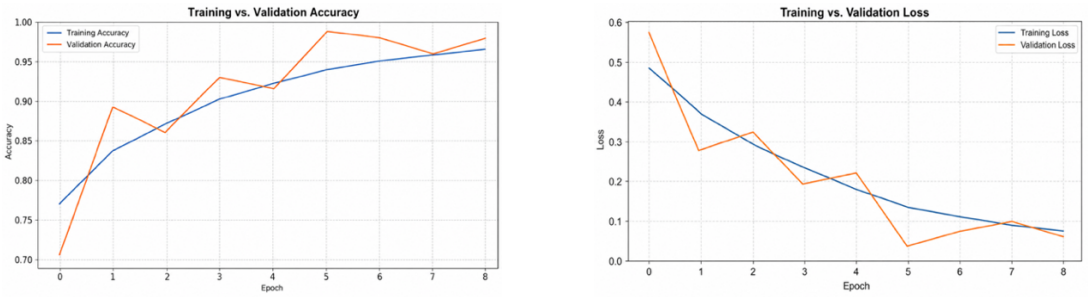
The model trained with SMOTE consistently outperformed the model trained without SMOTE across all evaluation metrics. Overall, the use of SMOTE led to improved accuracy and reduced loss, indicating better generalization and enhanced learning of minority class patterns, which is particularly important for detecting sexually harassing comments. The visualization of training and validation accuracy and loss indicates a tendency toward overfitting, as the gap between the training and validation curves increases with the number of epochs. Fig. 4 presents the training accuracy, training loss, validation accuracy, and validation loss for the model trained without SMOTE.



(a) (b)

Fig. 4 Curve of train vs validation set on LSTM without SMOTE: (a) accuracy amd (b) loss.

The visualizations for model with SMOTE are depicted Fig. 5.



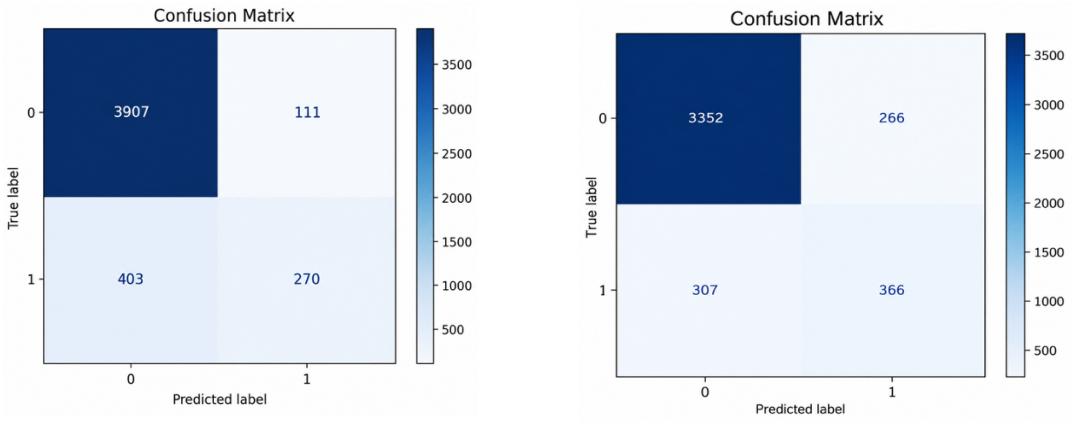
(a) (b)

Fig. 5 Word cloud of comments in both classes: (a) sexual harassment and (b) nonsexual harassment.

Additionally, the application of early stopping enabled the model to converge in fewer epochs, helping to prevent overfitting and reduce training time. Although SMOTE increased the processing time per step due to the larger training dataset, the resulting performance improvements justify this trade-off. Overall, these findings demonstrate that SMOTE effectively enhances the LSTM model's performance on imbalanced text classification tasks.

4.8. Evaluation Model

After comparing the graphs, model performance was further evaluated using a confusion matrix, which summarizes correct and incorrect predictions and facilitates assessment of classification effectiveness.



(a) (b)

Fig. 6 Confusion matrix of (a) LSTM without SMOTE and (b) LSTM+SMOTE.

As shown in Fig. 6(a), the model performed well in identifying non-harassment comments but failed to detect a substantial number of harassment instances, as indicated by the high number of false negatives. This pattern suggested a bias toward the majority class and limited generalization to minority class samples, further reflected by the relatively low number of true positives.

After applying SMOTE to balance the class distribution (Fig. 6(b)), the model demonstrated a marked improvement in recognizing harassment comments. The increase in true positives and reduction in false negatives indicated improved recall and stronger performance on the minority class. Although false positives increased slightly, this trade-off was acceptable given the model's enhanced ability to detect sexually harassing content.

The model's performance evaluation metrics were also calculated based on the results of the confusion matrix, specifically to measure the model's precision, recall, F1-score, and accuracy.

Table 10. Evaluation Metric of LSTM

Model	Class	Precision	Recall	F1-Score	Accuracy
LSTM	0	0.9065	0.9724	0.9383	0.8904
	1	0.7087	0.4012	0.5123	
LSTM+SMOTE	0	0.9244	0.9338	0.9244	0.8779
	1	0.5791	0.5438	0.5791	

Based on Table 10, applying SMOTE led to a modest improvement in recall for the harassment class, indicating better identification of minority class instances. However, this improvement was accompanied by a decrease in precision and overall accuracy. These results suggested that while SMOTE enhanced the model's sensitivity to harassment comments, it also increased misclassifications and slightly reduces overall predictive performance. A primary limitation is that SMOTE generates synthetic data through linear interpolation between numeric vectors without accounting for the complex semantic context captured by IndoBERT. Consequently, these 'synthetic'

vectors may exist as mathematical noise that does not correspond to realistic linguistic patterns, thereby confusing the LSTM and reducing overall predictive performance. Therefore, the use of SMOTE presents a trade-off between improved recall for the minority class and reduced precision and accuracy. Although SMOTE was applied to address data imbalance, the resulting model did not show a significant improvement in classifying sexual harassment comments. Therefore, this approach has not yet proven fully effective in handling data imbalance—particularly considering the main focus of this study is the identification of harassment-related content.

5. Conclusion

The word cloud visualization demonstrates a clear linguistic separation between harassment and non-harassment comments. Harassment-related content is dominated by explicit terms such as "*geol*," "*hyper*," "*tobrut*," and "*pulen*," which reflect patterns of physical objectification and sexual intent, whereas non-harassing comments primarily conveyed respectful appreciation without degrading expressions. The experimental results confirm that the LSTM model without SMOTE achieves strong performance on the majority class but remains insufficient in detecting harassment comments. Although the application of SMOTE improves minority class recall, it concurrently reduces precision and overall accuracy. These findings highlight that class imbalance handling through SMOTE introduces a measurable trade-off and does not inherently guarantee better model performance. Consequently, effective detection of online sexual harassment requires more than oversampling techniques, emphasizing the need for advanced modeling strategies or cost-sensitive approaches to improve minority class recognition without compromising overall accuracy.

Acknowledgment

The authors would like to express their sincere gratitude to the Statistics Laboratory for providing the facilities and software resources that supported the data analysis and the preparation of this manuscript.

References

[1] A. Thompson, "Digital 2026 global overview report," We Are Social Indonesia. Accessed: January 26, 2026. [Online]. Available: https://wearesocial.com/id/blog/2025/10/digital-2026-global-overview-report/

[2] S. Kemp, "Digital 2024 April global statshot report," DataReportal – Global Digital Insights. Accessed: January 26, 2026. [Online]. Available: https://datareportal.com/reports/digital-2024-april-global-statshot

[3] V.L. Kumar and M.A. Goldstein, "Cyberbullying and adolescents," *Curr. Pediatr. Rep.*, vol. 8, no. 3, pp. 86–92, Sep. 2020, doi: 10.1007/s40124-020-00217-6.

[4] G.W. Giumetti and R.M. Kowalski, "Cyberbullying via social media and well-being," *Curr. Opin. Psychol.*, vol. 45, Jun. 2022, Art. no 101314, doi: 10.1016/j.copsyc.2022.101314.

[5] "SIMFONI-PPA." Accessed: July 4, 2025. [Online]. Available: https://kekerasan.kemenpppa.go.id/ringkasan

[6] World Bank, "KBGO World Bank Report," 2023. [Online]. Available: https://documents.worldbank.org/en/publication/documents-reports/documentdetail/099637206212330304

[7] E.A. Vogels, "Online harassment occurs most often on social media, but strikes in other places, too," Pew Research Center. Accessed: July 4, 2025. [Online]. Available: https://www.pewresearch.org/short-reads/2021/02/16/online-harassment-occurs-most-often-on-social-media-but-strikes-in-other-places-too/

[8] G.Z. Nabiilah, I.N. Alam, E.S. Purwanto, and M.F. Hidayat, "Indonesian multilabel classification using IndoBERT embedding and MBERT classification," *Int. J. Electr. Comput. Eng.*, vol. 14, no. 1, pp. 1071–1078, Feb. 2024, doi: 10.11591/ijece.v14i1.pp1071-1078.

[9] G.H. Setiawan, M.D.A. Pranata, I.B.A. Arimbawa, I.W.P. Giri, and N.P.L.C. Dayani, "Topic clustering of student complaints based on semantic meaning using the indoBERT and k-means models," *J. Appl. Inform. Comput.*, vol. 9, no. 4, pp. 1715–1721, Aug. 2025, doi: 10.30871/jaic.v9i4.10080.

[10] T.C. Praha, W. Widodo, and M. Nugraheni, "Indonesian fake news classification using transfer learning in CNN and LSTM," *JOIV, Int. J. Inform. Vis.*, vol. 8, no. 3, pp. 1213–1221, Sep. 2024, doi: 10.62527/joiv.8.3.2126.

[11] D.Y. Yefferson, V. Lawijaya, and A.S. Girsang, "Hybrid model: IndoBERT and long short-term memory for detecting Indonesian hoax news," *Int. J. Artif. Intell.*, vol. 13, no. 2, pp. 1913–1924, Jun. 2024, doi: 10.11591/ijai.v13.i2.pp1913-1924.

[12] L. Khan, A. Amjad, K.M. Afaq, and H.-T. Chang, "Deep sentiment analysis using CNN-LSTM architecture of English and Roman Urdu text shared in social media," *Appl. Sci.*, vol. 12, no. 5, Jan. 2022, Art. no 2694, doi: 10.3390/app12052694.

[13] A.H. Primandari and P. Ermayani, "An empirical studies on online gender-based violence: Classification analysis utilizing XGBOOST," in *AIP Conf. Proc. 3248*, 2025, no. 1, Paper 040003, doi: 10.1063/5.0236703.

[14] V. Dogra et al., "A complete process of text classification system using state-of-the-art NLP models," *Comput. Intell. Neurosci.*, vol. 2022, 2022, Art. no 1883698, doi: 10.1155/2022/1883698.

[15] B. Wilie et al., "IndoNLU: Benchmark and resources for evaluating Indonesian natural language understanding," in *Proc. 1st Conf. Asia-Pac. Chapter Assoc. Comput. Ling. 10th Int. Joint Conf. Nat. Lang. Proc.*, s, 2020, pp. 843–857, doi: 10.18653/v1/2020.aacl-main.85.

[16] M.R. Farhan, A.A. Santoso, and J.J. Tedjasulaksana, "Sentiment analysis of public perception on freedom curriculum policy using text," in *2025 Int. Conf. Inf. Technol. Res. Innov. (ICITRI)*, 2025, pp. 1–6, doi: 10.1109/ICITRI67507.2025.11232776.

[17] A. Vaswani et al., "Attention is all you need," in *Adv. Neural Inf. Process. Syst.*, 2017, pp. 1–11.

[18] N.V. Chawla, K.W. Bowyer, L.O. Hall, and W.P. Kegelmeyer, "SMOTE: Synthetic minority over-sampling technique," *J. Artif. Intell. Res.*, vol. 16, pp. 321–357, Jun. 2002, doi: 10.1613/jair.953.

[19] S.F. Taskiran, B. Turkoglu, E. Kaya, and T. Asuroglu, "A comprehensive evaluation of oversampling techniques for enhancing text classification performance," *Sci. Rep.*, vol. 15, Jul. 2025, Art. no. 21631, doi: 10.1038/s41598-025-05791-7.

[20] A. Khurana and O P. Verma, "Optimal feature selection for imbalanced text classification," *IEEE Trans. Artif. Intell.*, vol. 4, pp. 135–147, Feb. 2023, doi: 10.1109/TAI.2022.3144651.

[21] A. Kaur, K.S. Gill, R. Chauhan, and H.S. Pokhariya, "Harnessing LSTM networks for enhanced text classification: A comprehensive study," in *2024 Int. Conf. Integr. Emerg. Technol. Digit. World (ICIETDW)*, 2024, pp. 1–5, doi: 10.1109/ICIETDW61607.2024.10939988.

[22] W.K. Sari, D.P. Rini, and R.F. Malik, "Text classification using long short-term memory," in *2019 Int. Conf. Electr. Eng. Comput. Sci. (ICECOS)*, 2019, pp. 150–155. doi: 10.1109/ICECOS47637.2019.8984558.

[23] Hermansah, M. Muhajir, and P.C. Rodrigues, "Indonesian Inflation Forecasting with Recurrent Neural Network Long Short-Term Memory (RNN-LSTM)," *Enthusiastic, Int. J. Appl. Stat. Data Sci.*, vol. 4, no. 2, pp. 132–142, Oct. 2024, doi: 10.20885/enthusiastic.vol4.iss2.art5.