Journal of Applied Informatics and Computing (JAIC)
Vol.10, No.1, February 2026, pp. 535~547
e-ISSN: 2548-6861

Optimizing Sentiment Classification Models for TikTok Comments using Emotion-Based Preprocessing and Grid Search

Bagas Restya Ermawan [1]*, Mahendra Bayu Prayoga [2]*, Akmal Rafi Fadhillah [3]*, Ema Utami [4]*
* Computer Engineering, University of Amikom Yogyakarta
b.restya@students.amikom.ac.id [1], m.bayu@students.amikom.ac.id [2], a.rafi@students.amikom.ac.id [3], ema.u@amikom.ac.id [4]

Article Info

Article history:

Received 2025-11-10
Revised 2025-12-25
Accepted 2026-01-07

Keyword:

Emotion-Based Preprocessing,
Grid Search,
Hyperparameter Tuning,
Machine Learning,
Sentiment Analysis.

ABSTRACT

TikTok has become one of the social media platforms with a significant influence on public opinion formation in Indonesia. However, the linguistic characteristics of user comments which are expressive, concise, and feature emotional forms like emojis, emoticons, and excessive capitalization pose challenges for sentiment analysis. This research aims to optimize a sentiment classification model for TikTok comments using emotion-based preprocessing and hyperparameter optimization via Grid Search. The dataset comprises 4,500 comments from three different time periods discussing the Minister of Finance, Purbaya Yudhi Sadewa. Three testing scenarios were conducted: common preprocessing, emotion-based preprocessing, and a combination of emotion-based preprocessing with Grid Search. The results indicate that emotion-based preprocessing improved model accuracy by 4–5%, while Grid Search optimization provided an additional increase of up to 3%, achieving a peak F1-score of 0.92 with the LightGBM model. Analysis based on sentiment time-periods reveals that across the three different periods, sentiments remained predominantly positive. The integration of emotion-based processing and parameter tuning proved effective in enhancing the model's ability to understand emotional variations in text and to map periodic changes in public sentiment on Indonesian-language social media.

I. Introduction

The development of information technology has had a significant influence on the transformation of communication patterns in modern society. Social media platforms like TikTok have now become new public spaces that allow users to openly express opinions, emotions, and perceptions regarding various social, political, and economic issues. Globally, TikTok has more than 100 million active users, and a large portion of its content generates thousands of comments that contain strong opinions and emotions [1]. These comments reflect the public's reaction to an event or figure, while also serving as a valuable data source for artificial intelligence-based sentiment analysis [2].

However, the linguistic characteristics of TikTok comments differ significantly from formal text. Users tend to write spontaneously, employing non-standard words (slang), abbreviations, emojis, emoticons, excessive capitalization, and expressive punctuation such as "!!!" or "???". These elements often carry strong emotional significance and can be crucial in determining sentiment polarity [3]. For example, the comment "GILA KEREN BANGET!!! □□" clearly indicates positive sentiment, yet traditional models that simply remove punctuation and emojis may lose this critical information. Consequently, traditional text preprocessing approaches can potentially degrade the accuracy of sentiment classification [4].

Previous research has highlighted the importance of Natural Language Processing (NLP) in sentiment analysis. Earlier studies generally focused on conventional text cleaning methods such as case folding, punctuation removal, tokenization, stopword removal, and stemming [5]. This approach is effective for formal texts, such as product reviews or news, but is not entirely suitable for social media text, which is laden with emotional context. Recent research confirms that the use of an emotion-based preprocessing approach can enhance the semantic representation of text by adding affective meaning from nonverbal expressions such as emojis, emoticons, and capitalization [6], [7].

Beyond the issue of preprocessing, the performance of sentiment classification models is also influenced by the selection and tuning of algorithm parameters. Most machine learning models, such as Support Vector Machine (SVM), Naïve Bayes, and Random Forest, possess hyperparameters that directly affect model performance. Suboptimal parameter selection can lead to model overfitting or underfitting, resulting in less accurate classification [8]. Hyperparameter tuning techniques, such as Grid Search, enable researchers to find the best parameter combination through a systematic evaluation of every possible configuration [9].

Research [10] showed that applying Grid Search to an SVM algorithm with N-Gram features improved model accuracy to 87.33% compared to the default configuration. However, limited research has combined emotion-based preprocessing with Grid Search in an integrated manner, particularly for Indonesian-language data on the TikTok platform. The majority of existing research has focused on other platforms, such as Twitter or YouTube. Furthermore, most studies have not systematically applied data balancing techniques like SMOTE (Synthetic Minority Over-sampling Technique) to address imbalanced sentiment distribution, even though applying SMOTE can enhance minority class representation and the performance stability of classification models [11].

This research seeks to address this gap by applying a combined approach of emotion-based preprocessing and Grid Search optimization to improve the accuracy of TikTok comment sentiment classification. The case research focuses on three critical moments involving the Minister of Finance, Purbaya Yudhi Sadewa, namely: the handover of duties from September 9–15, 2025, the one-month-in-office interview on October 10, 2025, and the unannounced inspection of hoarded used clothing in Cikarang on November 1, 2025. With a total of 4,500 comments, this research not only measures model performance but also maps the dynamics of public sentiment toward a government figure based on time and event context.

Unlike prior studies that independently apply emotion-aware preprocessing or hyperparameter optimization, this study emphasizes their systematic integration within the specific linguistic context of Indonesian TikTok comments. The novelty of this research lies not in proposing a new algorithm, but in constructing an emotion-sensitive preprocessing pipeline that explicitly captures TikTok-specific expressive features, such as excessive capitalization, repetitive punctuation, emojis, and emoticons, and evaluating its effectiveness when combined with Grid Search–optimized boosting models. Furthermore, this study extends prior sentiment classification research by incorporating a temporal sentiment analysis across distinct socio-political events, providing empirical insight into how public sentiment evolves in highly expressive social media environments.

It is expected that the results of this research will serve as an empirical foundation for the development of sentiment analysis methods for the Indonesian language, particularly within highly expressive social media environments. Practically, the proposed approach can support the development of AI-based public opinion monitoring systems capable of tracking sentiment dynamics across time. Academically, this research highlights the importance of integrating emotion-aware text processing with machine learning model optimization to produce more accurate and context-sensitive sentiment predictions.

II. METHODS

This research was conducted through several main stages, commencing with research design and culminating in model evaluation. Data obtained during the dataset collection phase was first processed using a series of text preprocessing steps, such as data cleaning, case folding, text normalization, tokenizing, stopword removal, and stemming. Subsequently, the data was assigned sentiment labels using a lexicon-based approach, and features were extracted using the TF-IDF method. The data was then divided into training and testing sets, with the SMOTE technique applied to address class imbalance. Finally, the sentiment classification model was developed and optimized using Grid Search, then evaluated to measure its performance and classification accuracy.

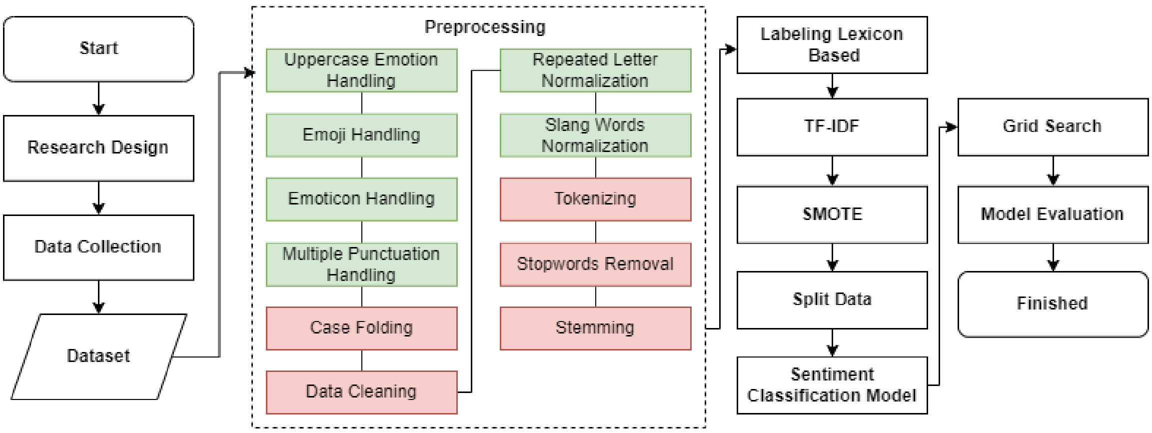

Figure 1. Research Flow

A. Research Design

A quantitative experimental approach was employed in this research to measure the impact of applying emotion-based preprocessing and hyperparameter optimization on the sentiment classification performance of TikTok comments. The research process was divided into two experimental stages: the first experiment utilized a conventional preprocessing approach without emotion-based handling, and the second experiment implemented emotion-based preprocessing optimized using the Grid Search method. The comparison between these two experiments aims to assess the extent to which emotion-based treatment and parameter tuning can enhance model accuracy.

The research stages included TikTok comment dataset collection, text preprocessing, labeling, feature extraction using TF-IDF, class distribution balancing using the SMOTE method, splitting the data into training and testing subsets, training the sentiment classification model, parameter optimization using Grid Search, and evaluating model performance based on accuracy, precision, recall, and F1-score metrics. All experiments were implemented in the Python programming language on the Google Colab platform, utilizing the core libraries pandas, NumPy, scikit-learn, NLTK, Sastrawi, and emoji.

B. Data Collection

This research utilizes data collected from TikTok user comments on several news accounts that featured coverage of the Minister of Finance, Purbaya Yudhi Sadewa. The collection process was performed manually using the Apify tool [12]. Data was sampled by collecting 1,500 comments from each of three videos representing different event contexts, resulting in a total of 4,500 comment data points. These three videos included: the handover of duties, uploaded on September 9, 2025; the one-month-in-office interview, uploaded on October 10, 2025; and the unannounced inspection of hoarded used clothing in Cikarang, uploaded on November 1, 2025. The selection of these three moments was based on their high comment activity and relevance to public opinion regarding the policies and image of the government figure.

The selection of the three observation periods was guided by their distinct socio-political contexts, each representing a different phase of public perception. The first period captures initial public reactions following the handover of office, typically characterized by optimism and expectation. The second period reflects evaluative sentiment after one month of leadership performance, during which public scrutiny intensifies. The third period corresponds to a concrete policy-related action, which often elicits stronger emotional responses, including support and criticism. This temporal segmentation enables a more meaningful interpretation of sentiment dynamics rather than treating public opinion as static.

From each comment collected, two main columns were extracted: createTimeISO, representing the comment upload time, and text, which contains the textual content of the comment. The dataset was then saved in CSV format under the filename purbaya_text.csv as the primary data for the analysis process. Before the preprocessing stage, an initial cleaning was applied, involving the removal of duplicate data and null values, leaving 4,485 valid data points ready to be processed in the next stage. The dataset can be seen in Table I.

TABLE I
DATASET

id	createTimeISO	Text
1	2025-10-10T02:06:31.000Z	Begini kl mentri non parpol, ngomongnya loss aja krn ga ada tekanan □
2	2025-10-10T05:15:38.000Z	semoga semua menteri bisa berkaca pada Pak Purbaya krn sdh menjadi idola seluruh rakyat□□
3	2025-10-10T02:03:32.000Z	percayakan saja sma pak pur□□
…	…	…
4485	2025-11-01T14:37:52.000Z	anak buah kemenkeu dari raut muka nggak bisa dipercaya. Pecat saja, pak Purbaya. Kami rakyat mendukungmu

C. Preprocessing

Emotion-based preprocessing in this study is implemented using a rule-based emotion encoding strategy to preserve emotional intensity commonly found in TikTok comments. Capitalization-based emotion encoding is applied by detecting comments written entirely or predominantly in uppercase, which are then marked by appending a special token (token_uppercase) at the end of the text. Emoji symbols are handled using a JSON-based emoji dictionary, where each emoji is converted into a corresponding emotion-related word. In contrast, Unicode emoticons are processed using the Python emoji library and similarly transformed into textual emotion representations. Expressive punctuation patterns, such as repeated exclamation or question marks, are replaced with a dedicated emphasis marker (token_penekanan).

Additionally, words containing repeated letters are normalized to their base forms to reduce lexical sparsity, while simultaneously being annotated with the token_penekanan marker to retain their expressive emphasis. All emotion-related tokens are treated as standard lexical features and are numerically represented through the TF-IDF vectorization process, allowing emotional signals to influence the feature space implicitly based on their frequency and discriminative power within the corpus, without applying explicit or arbitrary weighting schemes.

The preprocessing stage was performed to transform raw comment data into clean and structured text before proceeding to the feature extraction stage. The entire process was implemented using Python within the Google Colab environment, utilizing the pandas, re, NLTK, Sastrawi, emoji, and json libraries. The sequence of preprocessing steps applied in this research is detailed as follows.

1) *Uppercase Emotion Handling*

Comments written entirely in capital letters are considered to express strong emotions such as anger, admiration, or enthusiasm [13]. To retain this information, the system appends a special token, 'token_uppercase', to texts detected as fully capitalized. This is shown in Table II.

TABLE II
RESULT OF UPPERCASE EMOTION HANDLING

Text	Uppercase Emotion Handling
SALUUUUUUTTTT.. SAMA BPK INI.. GA MAU DI ATUR ATUR SAMA KEPENTINGAN KELOMPOK.. IS THE BEST UNTUK RAKYAT □□□□□	SALUUUUUUTTTT.. SAMA BPK INI.. GA MAU DI ATUR ATUR SAMA KEPENTINGAN KELOMPOK.. IS THE BEST UNTUK RAKYAT. □□□□□ token_uppercase
"Media suka mancing-mancing yg ngga keruan..!! ""SALUUUUTTT DGN JAWABAN PAK PURBAYA..!!"""	"Media suka mancing-mancing yg ngga keruan..!! ""SALUUUUTTT DGN JAWABAN PAK PURBAYA..!!"""
"justru tugas menteri perdagangan itu membina produk"" dlm negeri agar lebih berkwalitas, Gaasss poooolll Purbaya, kasihan para pekerja garment gajinya cm pas"" an" :)	"justru tugas menteri perdagangan itu membina produk"" dlm negeri agar lebih berkwalitas, Gaasss poooolll Purbaya, kasihan para pekerja garment gajinya cm pas"" an" :)

2) *Emoji Handling*

Text symbols such as :), :(, :D, and ;) are recognized using an emoji dictionary based on a JSON file. Emoji replacement can enhance the model's ability to understand sentence context and detect sarcasm [14]. Each emoji is replaced with a relevant emotion word, as shown in the example in Table III.

TABLE III
RESULT OF EMOJI HANDLING

Uppercase Emotion Handling	Emoji Handling
SALUUUUUUTTTT.. SAMA BPK INI.. GA MAU DI ATUR ATUR SAMA KEPENTINGAN KELOMPOK.. IS THE BEST UNTUK RAKYAT. □□□□□ token_uppercase	SALUUUUUUTTTT.. SAMA BPK INI.. GA MAU DI ATUR ATUR SAMA KEPENTINGAN KELOMPOK.. IS THE BEST UNTUK RAKYAT. □□□□□ token_uppercase
"Media suka mancing-mancing yg ngga keruan..!! ""SALUUUUTTT DGN JAWABAN PAK PURBAYA..!!"""	"Media suka mancing-mancing yg ngga keruan..!! ""SALUUUUTTT DGN JAWABAN PAK PURBAYA..!!"""
"justru tugas menteri perdagangan itu membina produk"" dlm negeri agar lebih berkwalitas, Gaasss poooolll Purbaya, kasihan para pekerja garment gajinya cm pas"" an" :)	"justru tugas menteri perdagangan itu membina produk"" dlm negeri agar lebih berkwalitas, Gaasss poooolll Purbaya, kasihan para pekerja garment gajinya cm pas"" an" senyum

3) *Emoticon Handling*

Unicode emoticons such as □, □, or □ are converted into their word equivalents using the *emoji* library in Python. In addition to emoji replacement, emoticon replacement can also enhance the model's ability to understand sentence context and detect sarcasm [14]. As shown in the example in Table IV.

TABLE IV
RESULT OF UPPERCASE EMOTICON HANDLING

Emoji Handling	Emoticon Handling
SALUUUUUUTTTT.. SAMA BPK INI.. GA MAU DI ATUR ATUR SAMA KEPENTINGAN KELOMPOK.. IS THE BEST UNTUK RAKYAT. □□□□□ token_uppercase	SALUUUUUUTTTT.. SAMA BPK INI.. GA MAU DI ATUR ATUR SAMA KEPENTINGAN KELOMPOK.. IS THE BEST UNTUK RAKYAT. jempol ke atas hati merah token_uppercase
"Media suka mancing-mancing yg ngga keruan..!! ""SALUUUUTTT DGN JAWABAN PAK PURBAYA..!!"""	"Media suka mancing-mancing yg ngga keruan..!! ""SALUUUUTTT DGN JAWABAN PAK PURBAYA..!!"""
"justru tugas menteri perdagangan itu membina produk"" dlm negeri agar lebih berkwalitas, Gaasss poooolll Purbaya, kasihan para pekerja garment gajinya cm pas"" an" senyum	"justru tugas menteri perdagangan itu membina produk"" dlm negeri agar lebih berkwalitas, Gaasss poooolll Purbaya, kasihan para pekerja garment gajinya cm pas"" an" senyum

4) *Multiple Expressive Handling*

While proper punctuation usage can convey meaning, excessive repetition requires careful handling during sentiment analysis [15]. Repetitive punctuation patterns, such as !!!, are considered to indicate emphasis or heightened emotion. These patterns are replaced with the token 'token_penekanan' to allow the model to distinguish between standard comments and those with stronger emotional intensity, as exemplified in Table V.

TABLE V
RESULT OF MULTIPLE EXPRESSIVE HANDLING

Emoticon Handling	Multiple Expressive Handling
SALUUUUUUTTTT.. SAMA BPK INI.. GA MAU DI ATUR ATUR SAMA KEPENTINGAN KELOMPOK.. IS THE BEST UNTUK RAKYAT. jempol ke atas hati merah token_uppercase	SALUUUUUUTTTT.. SAMA BPK INI.. GA MAU DI ATUR ATUR SAMA KEPENTINGAN KELOMPOK.. IS THE BEST UNTUK RAKYAT. jempol ke atas hati merah token_uppercase
"Media suka mancing-mancing yg ngga keruan..!! ""SALUUUUTTT DGN JAWABAN PAK PURBAYA..!!"""	"Media suka mancing-mancing yg ngga keruan.. ""SALUUUUTTT DGN JAWABAN PAK PURBAYA.. """ token_penekanan
"justru tugas menteri perdagangan itu membina produk"" dlm negeri agar lebih berkwalitas, Gaasss poooolll Purbaya, kasihan para pekerja garment gajinya cm pas"" an" senyum	"justru tugas menteri perdagangan itu membina produk"" dlm negeri agar lebih berkwalitas, Gaasss poooolll Purbaya, kasihan para pekerja garment gajinya cm pas"" an" senyum

5) *Case Folding*

All letters are converted to lowercase to standardize word forms and avoid discrepancies in meaning due to capitalization [16]. An example can be seen in Table VI.

TABLE VI
RESULT OF CASE FOLDING

Multiple Expressive	Case Folding
SALUUUUUUTTTT.. SAMA BPK INI.. GA MAU DI ATUR ATUR SAMA KEPENTINGAN KELOMPOK.. IS THE BEST UNTUK RAKYAT. jempol ke atas hati merah token_uppercase	saluuuuuutttt.. sama bpk ini.. ga mau di atur atur sama kepentingan kelompok.. is the best untuk rakyat. jempol ke atas hati merah token_uppercase
"Media suka mancing-mancing yg ngga keruan.. ""SALUUUUTTT DGN JAWABAN PAK PURBAYA.. """ token_penekanan	"media suka mancing-mancing yg ngga keruan.. ""saluuuuttt dgn jawaban pak purbaya.. """ token_penekanan
"justru tugas menteri perdagangan itu membina produk"" dlm negeri agar lebih berkwalitas, Gaasss poooolll Purbaya, kasihan para pekerja garment gajinya cm pas"" an" senyum	"justru tugas menteri perdagangan itu membina produk"" dlm negeri agar lebih berkwalitas, gaasss poooolll purbaya, kasihan para pekerja garment gajinya cm pas"" an" senyum

6) *Cleaning*

Cleaning is performed to remove non-text elements such as URLs, mentions (@username), hashtags (#topic), numbers, and punctuation that lack significant semantic value [14]. This process utilizes regular expressions (regex) to ensure text consistency, as shown in Table VII.

TABLE VII
RESULT OF CLEANING

Case Folding	Cleaning
saluuuuuutttt.. sama bpk ini.. ga mau di atur atur sama kepentingan kelompok.. is the best untuk rakyat. jempol ke atas hati merah token_uppercase	saluuuuuutttt sama bpk ini ga mau di atur atur sama kepentingan kelompok is the best untuk rakyat jempol ke atas hati merah token_uppercase
"media suka mancing-mancing yg ngga keruan.. ""saluuuuttt dgn jawaban pak purbaya.. """ token_penekanan	media suka mancing mancing yg ngga keruan saluuuuttt dgn jawaban pak purbaya token_penekanan
"justru tugas menteri perdagangan itu membina produk"" dlm negeri agar lebih berkwalitas, gaasss poooolll purbaya, kasihan para pekerja garment gajinya cm pas"" an" senyum	justru tugas menteri perdagangan itu membina produk dlm negeri agar lebih berkwalitas gaasss poooolll purbaya kasihan para pekerja garment gajinya cm pas an senyum

7) *Repeated Letter Handling*

Characters repeated more than twice, such as in "baguuuuus" or "paraaaah", are reverted to their normal form (e.g., "bagus" and "parah"). This process prevents tokenization bias caused by expressive spellings common on social media [17]. An example can be seen in Table VIII.

TABLE VIII
RESULT OF REPEATED LETTER HANDLING

Cleaning	Repeated Letter Handling
saluuuuuutttt sama bpk ini ga mau di atur atur sama kepentingan kelompok is the best untuk rakyat. jempol ke atas hati merah token_uppercase	salut sama bpk ini ga mau di atur atur sama kepentingan kelompok is the best untuk rakyat jempol ke atas hati merah token_uppercase token_penekanan
media suka mancing mancing yg ngga keruan saluuuuttt dgn jawaban pak purbaya token_penekanan	media suka mancing mancing yg ngga keruan salut dgn jawaban pak purbaya token_penekanan
justru tugas menteri perdagangan itu membina produk dlm negeri agar lebih berkwalitas gaasss poooolll purbaya kasihan para pekerja garment gajinya cm pas an senyum	justru tugas menteri perdagangan itu membina produk dlm negeri agar lebih berkwalitas gaas pol purbaya kasihan para pekerja garment gajinya cm pas an senyum token_penekanan

8) *Slangword Normalization*

Normalization of non-standard words is performed to standardize the written form of words that are commonly used informally [18]. These word forms are frequently encountered on social media, such as "bgt" being normalized to "banget" or "ngga" to "tidak". This process utilizes a slang word dictionary compiled from four GitHub repositories, which collectively contain 5,433 non-standard and standard word pairs. These four dictionaries were merged into a single JSON-formatted file and subsequently converted into a file named fix_slangwords.txt to serve as the primary reference for the normalization process. An example is provided in Table IX.

TABLE IX
RESULT OF SLANGWORD NORMALIZATION

Repeated Letter Handling	Slangword Normalization
salut sama bpk ini ga mau di atur atur sama kepentingan kelompok is the best untuk rakyat jempol ke atas hati merah token_uppercase token_penekanan	salut sama bapak ini tidak mau di atur atur sama kepentingan kelompok is the best untuk rakyat jempol ke atas hati merah token_uppercase token_penekanan
media suka mancing mancing yg ngga keruan saluuuuttt dgn jawaban pak purbaya token_penekanan	media suka mancing mancing yang tidak keruan salut dengan jawaban pak purbaya token_penekanan
justru tugas menteri perdagangan itu membina produk dlm negeri agar lebih berkwalitas gaas pol purbaya kasihan para pekerja garment gajinya cm pas an senyum token_penekanan	justru tugas menteri perdagangan itu membina produk dalam negeri agar lebih berkualitas gaas pol purbaya kasihan para pekerja garment gajinya cuma cocok an senyum token_penekanan

9) *Tokenizing*

Tokenization is the process of breaking down text into individual tokens such as words, numbers, punctuation, or other special symbols by determining the end point of one word and the starting point of the next, which is known as a word boundary [15]. This process simplifies the subsequent stages of stopword removal and stemming, as shown in Table X.

TABLE X
RESULT OF TOKENIZING

Slangword Normalization	Tokenizing
salut sama bapak ini tidak mau di atur atur sama kepentingan kelompok is the best untuk rakyat jempol ke atas hati merah token_uppercase token_penekanan	['salut', 'sama', 'bapak', 'ini', 'tidak', 'mau', 'di', 'atur', 'atur', 'sama', 'kepentingan', 'kelompok', 'is', 'the', 'best', 'untuk', 'rakyat', 'token_uppercase', 'token_penekanan',]
media suka mancing mancing yang tidak keruan salut dengan jawaban pak purbaya token_penekanan	['media', 'suka', 'mancing', 'mancing', 'yang', 'tidak', 'keruan', 'salut', 'dengan', 'jawaban', 'pak', 'purbaya', 'token_penekanan']
justru tugas menteri perdagangan itu membina produk dalam negeri agar lebih berkualitas gaas pol purbaya kasihan para pekerja garment gajinya cuma cocok an senyum token_penekanan	['justru', 'tugas', 'menteri', 'perdagangan', 'itu', 'membina', 'produk', 'dalam', 'negeri', 'agar', 'lebih', 'berkualitas', 'gaas', 'pol', 'purbaya', 'kasihan', 'para', 'pekerja', 'garment', 'gajinya', 'cuma', 'cocok', 'an', 'senyum', 'token_penekanan']

10) Stopword Removal

Stopword Removal is the process of eliminating words, such as articles and pronouns, that appear with high frequency throughout a text and are irrelevant to the task being performed [15]. Common words that do not carry significant meaning, such as “yang”, “dan”, “di”, and “ke”, are removed by utilizing the Indonesian stopwords list provided by the NLTK library, as shown in Table XI.

TABLE XI
RESULT OF STOPWORD REMOVAL

Repeated Letter Handling	Slangword Normalization
['saluttttt', 'token_penekanan', 'maun', 'atur', 'atur', 'kepentingan', 'kelompok', 'is', 'the', 'best', 'rakyat', 'token_uppercase']	['salut', 'mau', 'atur', 'atur', 'kepentingan', 'kelompok', 'is', 'the', 'best', 'rakyat', 'token_uppercase', 'token_penekanan']
['media', 'suka', 'mancing', 'mancing', 'yang', 'tidak', 'keruan', 'salut', 'dengan', 'jawaban', 'pak', 'purbaya', 'token_penekanan']	['media', 'suka', 'mancing', 'mancing', 'keruan', 'salut', 'purbaya', 'token_penekanan']
['justru', 'tugas', 'menteri', 'perdagangan', 'itu', 'membina', 'produk', 'dalam', 'negeri', 'agar', 'lebih', 'berkualitas', 'gaas', 'pol', 'purbaya', 'kasihan', 'para', 'pekerja', 'garment', 'gajinya', 'cuma', 'cocok', 'an', 'senyum', 'token_penekanan']	['tugas', 'menteri', 'perdagangan', 'membina', 'produk', 'negeri', 'berkualitas', 'gaas', 'pol', 'purbaya', 'kasihan', 'pekerja', 'garment', 'gajinya', 'cocok', 'an', 'senyum', 'token_penekanan']

11) Stemming

Each token is reduced to its base form using the Sastrawi Stemmer. For example, the words “berlarian”, “lari-lari”, and “berlari” are all standardized to “lari”. Thus, stemming helps consolidate various variations of a token into a single entity [15]. This process improves the efficiency and consistency of the text features, as shown in Table XII.

TABLE XII
RESULT OF STEMMING

Stopword Removal	Stemming
['salut', 'mau', 'atur', 'atur', 'kepentingan', 'kelompok', 'is', 'the', 'best', 'rakyat', 'token_uppercase', 'token_penekanan']	['salut','mau', 'atur', 'atur', 'penting', 'kelompok', 'is', 'the', 'best', 'rakyat', 'token_uppercase', 'token_penekanan']
['media', 'suka', 'mancing', 'mancing', 'keruan', 'salut', 'purbaya', 'token_penekanan']	['media', 'suka', 'mancing', 'mancing', 'keruan', 'salut', 'purbaya', 'token_penekanan']
['tugas', 'menteri', 'perdagangan', 'membina', 'produk', 'negeri', 'berkualitas', 'gaas', 'pol', 'purbaya', 'kasihan', 'pekerja', 'garment', 'gajinya', 'cocok', 'an', 'senyum', 'token_penekanan']	['tugas', 'menteri', 'dagang', 'bina', 'produk', 'negeri', 'kualitas', 'gaas', 'pol', 'purbaya', 'kasihan', 'kerja', 'garment', 'gaji', 'cocok', 'an', 'senyum', 'token_penekanan']

After all preprocessing stages were completed, the final results were stored in a new column named text_preprocessed, which contains the clean text ready for the feature extraction stage.

TABLE XIII
RESULT OF PREPROCESSING

Stemming	Text_Preprocessed
['salut','mau', 'atur', 'atur', 'penting', 'kelompok', 'is', 'the', 'best', 'rakyat', 'token_uppercase', 'token_penekanan']	salut maun atur atur penting kelompok is the best rakyat token_uppercase token_penekanan
['media', 'suka', 'mancing', 'mancing', 'keruan', 'salut', 'purbaya', 'token_penekanan']	media suka mancing mancing keruan salut purbaya token_penekanan
['tugas', 'menteri', 'dagang', 'bina', 'produk', 'negeri', 'kualitas', 'gaas', 'pol', 'purbaya', 'kasihan', 'kerja', 'garment', 'gaji', 'cocok', 'an', 'senyum', 'token_penekanan']	tugas menteri dagang bina produk negeri kualitas gaas pol purbaya kasihan kerja garment gaji cocok an senyum token_penekanan

D. Labeling Lexicon-Based

The labeling stage was conducted after the preprocessing process using a lexicon-based approach, a method that determines sentiment polarity based on the scores of positive and negative words [19]. Two sentiment lexicons from GitHub repositories were used as references: the first lexicon contained 3,609 positive and 6,609 negative words, while the second covered 1,729 words with mixed sentiment scores. To enrich the vocabulary and ensure the consistency of polarity values, the two lexicons were merged by removing 842 duplicate words.

The merging process resulted in two final lexicons: positive.txt, containing 3,873 positive words, and negative.txt, containing 7,230 negative words. Each comment was then assigned a sentiment score based on the summation of values from each word in the text, with scores ranging from –5 to +5. Comments with a total score => 0 were classified as positive, while scores < 0 were classified as negative.

Out of the total 4,485 comments analyzed, the labeling results indicated that positive sentiment comments outnumbered negative ones. The distribution of the labeling results can be seen in the following Figure 2.

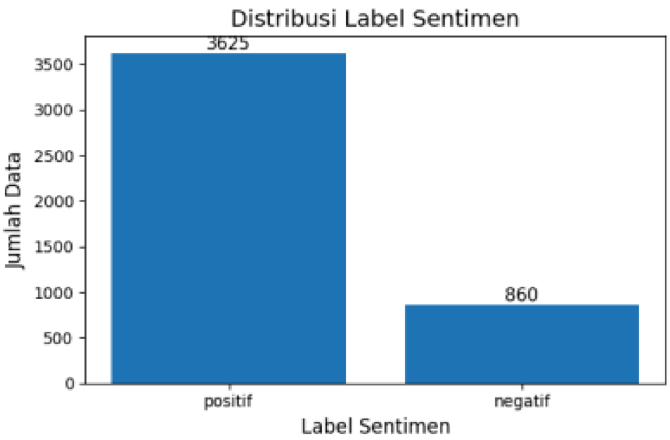

Figure 2. Distribution of Sentiment Labeling Results

From Figure 2, it is evident that the sentiment distribution consists of 3,625 positive comments (approximately 81%) and 860 negative comments (approximately 19%). This imbalance indicates that positive sentiment dominated the public discourse surrounding the analyzed topic. Such a disproportionate class distribution may bias classification models toward the majority class and potentially inflate performance metrics if not properly addressed. Therefore, the SMOTE technique was applied in the subsequent stage to balance the class distribution prior to model training, ensuring a more reliable and robust evaluation of sentiment classification performance.

E. TF-IDF

The feature extraction stage was conducted using the Term Frequency–Inverse Document Frequency (TF-IDF) method to represent the preprocessed text in a numerical format. TF-IDF calculates the importance of a term to a document based on its occurrence frequency in a specific comment compared to the entire set of comments (corpus) [20]. Terms that appear frequently in one comment but infrequently in others are assigned a higher weight and are deemed more representative of the sentiment context. From the TF-IDF calculation results on the TikTok comment dataset, the ten terms with the highest weights were identified, as shown in the following Table XIV.

TABLE XIV
TOP 10 HIGHEST TF-IDF VALUE

Word	TF-IDF Value
menkeu	0.7231
bicara	0.7071
cawapres	0.6658
atur	0.5774
pur	0.5774
percaya	0.4637
ekonomi	0.4637
beliau	0.4637
fokus	0.4637
kait	0.4637

F. SMOTE

SMOTE (Synthetic Minority Oversampling Technique) was used to balance the amount of data between the positive and negative classes. This technique generates new synthetic samples from the minority class through interpolation between adjacent data points in the feature space, thereby balancing the data distribution and preventing the model from being biased toward the majority class [21].

After the application of SMOTE, the number of positive and negative data became equal, each totaling 3,625 data points. The distribution of the oversampling results is shown in Figure 3, whereas the distribution before SMOTE can be seen in the previous Figure 2.

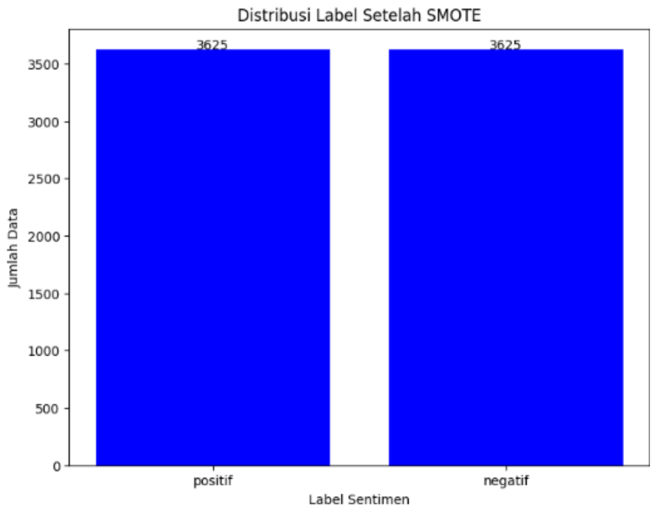

Figure 3. Label Distribution After SMOTE

G. Split Data

A common ratio is 80% for training and 20% for testing, although this proportion is subject to variation [22]. In this research, the dataset was divided into two subsets: an 80% training set and a 20% testing set. The split was performed randomly to avoid bias and ensure model generalization. The training set was used to build the classification model, while the testing set served to evaluate the model's performance on new, previously unseen data.

H. Sentiment Classification Model

This research employs three main models to classify the sentiment of the preprocessed text data: Gradient Boosting, XGBoost, and LightGBM. These three models were selected because they belong to the family of ensemble boosting algorithms.

1) *Gradient Boosting*

Gradient Boosting was utilized as a baseline model to observe the boosting algorithm's capability of correcting prediction errors from the previous model in the sequence. This model constructs decision trees sequentially, where each iteration focuses on data that was previously misclassified

[23]. With this approach, the model is expected to capture subtler sentiment patterns, including comments with sarcastic or expressive tones.

2) XGBoost

In this research, XGBoost was selected because it possesses strong regularization capabilities for preventing overfitting that can result from imbalanced training data, even after balancing with SMOTE. This model is also capable of accelerating the training process within the Google Colab environment through its support for parallelization, making it suitable for processing a large volume of comments [23].

3) LightGBM

LightGBM was utilized to test the performance of a boosting model offering higher computational efficiency. LightGBM operates using a leaf-wise growth strategy, which enables the model to handle variations in text length and sentence structure complexity more adaptively [23]. In this research, LightGBM was expected to deliver the best classification results for social media comments, which possess a linguistic style that is concise yet dense with emotional meaning.

I. Grid Search

Hyperparameter Tuning was performed using the Grid Search Cross Validation (GridSearchCV) method from the scikit-learn library to determine the best parameter combination for the three boosting models: Gradient Boosting, XGBoost, and LightGBM. This process evaluated various parameter combinations through 5-fold cross validation and selected the best-performing configuration based on the F1-score [24]. The parameters optimized for each model included n_estimators, learning_rate, and max_depth for Gradient Boosting; n_estimators, learning_rate, max_depth, and colsample_bytree for XGBoost; and n_estimators, learning_rate, max_depth, and num_leaves for LightGBM. This approach ensures that the models are not only optimal on the training data but also possess good generalization capabilities for the testing data.

J. Evaluation

Model evaluation was performed to measure sentiment classification performance using four main metrics: accuracy, precision, recall, and F1-score [14]. Accuracy indicates the proportion of correct predictions relative to the entire test data (1), precision measures the correctness of the model's positive predictions (2), while recall indicates the model's ability to identify all actual positive instances (3). The F1-score is used as the harmonic mean of precision and recall to provide a balanced assessment (4), particularly on datasets with imbalanced class distributions. These four metrics are calculated using the following equations.

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

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

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

$$F1 = 2 \times \frac{(Precision \times Recall)}{(Precision + Recall)} \tag{4}$$

In addition to numerical metrics, the prediction results are also visualized using a confusion matrix to illustrate the distribution of correct and incorrect predictions for each sentiment class. The model achieving the highest F1-score following Grid Search optimization is deemed to have the best performance and is selected as the final model for this research.

III. RESULTS AND DISCUSSION

A. Comparison of Preprocessing Scenarios Without and With the Addition of Emotion-Based

Experiments were conducted using two text preprocessing approaches, conventional preprocessing (without emotion-based handling) and preprocessing with the addition of an emotion-based stage. The three boosting models Gradient Boosting (GB), XGBoost (XGB), and LightGBM (LGBM) were evaluated using an 80:20 split between the training and testing sets. This ratio was selected to ensure sufficient data for model training while maintaining an independent testing set for reliable evaluation of model generalization performance, a practice widely adopted in text classification research. The results obtained using conventional preprocessing are presented in Table XV.

TABLE XV
MODEL RESULTS WITH GENERAL PREPROCESSING

Models	Accuracy	Precision	Recall	F1-Score
GB	85%	87%	85%	85%
XGB	87%	88%	87%	87%
LGB	85%	86%	85%	85%

Next, the results with emotion-based preprocessing are shown in Table XVI below.

TABLE XVI
MODEL RESULTS WITH EMOTION-BASED PREPROCESSING

Models	Accuracy	Precision	Recall	F1-Score
GB	85%	86%	85%	85%
XGB	90%	91%	90%	90%
LGB	92%	92%	92%	92%

The results indicate that the addition of emotion-based preprocessing led to performance improvements in the XGBoost and LightGBM models, while the Gradient Boosting model exhibited relatively stable performance across both preprocessing scenarios. The LightGBM model demonstrated the most pronounced performance improvement, with its F1-score increasing from 0.85 to 0.92 following the application of emotion-based preprocessing. This improvement is attributable to LightGBM's leaf-wise tree growth strategy, which exhibits higher sensitivity to informative and sparse features compared to level-wise boosting approaches. The emotion-based preprocessing stage introduced explicit emotional tokens derived from capitalization patterns, emojis, emoticons, and repetitive punctuation, thereby enhancing the discriminative capacity of sparse TF-IDF feature representations. Consequently, LightGBM was able to more effectively leverage these emotion-enriched features to identify nuanced sentiment signals that are characteristic of highly expressive TikTok user comments.

B. Comparison of Emotion-Based Preprocessing Scenarios with and without Grid Search

This stage compares the sentiment classification results using emotion-based preprocessing before and after hyperparameter optimization with the Grid Search Cross Validation (GridSearchCV) method. The performance values before optimization (without Grid Search) refer to the emotion-based preprocessing results that were presented in Table XVI. Meanwhile, the results after optimization with Grid Search are shown in the following Table XVII.

TABLE XVII
MODEL RESULTS WITH EMOTION-BASED GRID SEARCH

Models	Accuracy	Precision	Recall	F1-Score
GB	91%	91%	91%	91%
XGB	91%	91%	91%	91%
LGB	92%	92%	92%	92%

The optimization process was conducted to determine the best parameter combination for each model. Each model had different parameters tested and distinct optimal values, as shown in Table XVIII through Table XX.

TABLE XVIII
GRADIENT BOOSTING PARAMETERS

Parameters	Tested Values	Optimal Value
n_estimators	[50,100, 150]	150
learning_rate	[0.05, 0.1, 0.2]	0.2
max_depth	[3, 5, 8]	2

TABLE XIX
XGBOOST PARAMETERS

Parameters	Tested Values	Optimal Value
n_estimators	[50,100, 150]	150
learning_rate	[0.05, 0.1, 0.2]	0.2
max_depth	[3, 5, 8]	8
colsample_bytree	[0.7, 1.0]	1.0

TABLE XX
LIGHTGBM PARAMETERS

Parameters	Tested Values	Optimal Value
n_estimators	[50,100, 150]	100
learning_rate	[0.05, 0.1, 0.2]	0.2
max_depth	[-1, 10, 20]	-1
colsample_bytree	[20, 31, 40]	31

Each model demonstrated a different parameter configuration to achieve optimal performance. For Gradient Boosting, the combination of a learning_rate of 0.2, a max_depth of 5, and 150 n_estimators yielded a balance between learning speed and model complexity. The XGBoost model obtained its best results with a similar combination, but with the addition of colsample_bytree at 1.0, which broadened the feature scope in each decision tree. Meanwhile, LightGBM achieved optimal results with 31 num_leaves, a learning_rate of 0.2, a max_depth of -1 (indicating no depth limit), and 100 estimators, allowing for the formation of deeper and more efficient trees capable of capturing complex sentiment patterns.

C. Comparison Analysis of General Preprocessing and Emotion-Based Preprocessing

The research results demonstrate that the application of emotion-based preprocessing yields a significant performance improvement compared to conventional preprocessing. This enhancement occurred consistently across all three boosting models used: Gradient Boosting, XGBoost, and LightGBM. Notably, the LightGBM model exhibited the most prominent increase, with its F1-score rising from 0.85 to 0.92.

This improvement indicates that the emotion-based preprocessing stage successfully captures the emotional expressions conveyed through capitalization, emojis, and emoticons within TikTok user comments [6]. Converting these emotional elements into semantically meaningful text forms significantly contributes to the enhancement of sentiment analysis accuracy. Models lacking emotion-based preprocessing tend to struggle with distinguishing between strongly positive comments and merely expressive neutral comments [14]. For example, a comment like "BAGUS BANGET!!! □" would lose its emotional meaning if it only undergoes conventional preprocessing.

Through an emotion-based approach, non-lexical information such as capitalization and expressive symbols can be converted into features with semantic weight, thereby enhancing the model's ability to capture emotional context more accurately [15]. Previous research has also shown that the integration of emotional elements, like emojis and non-lexical symbols, can improve the performance of sentiment analysis models, particularly when combined with deep learning models such as BERT [4]. Furthermore, the handling of repetitive punctuation, or multiple expressive handling, further assists the model in understanding the emotional intensity implied in user comments [15].

D. Effectiveness of Grid Search Application in Model Optimization

The optimization stage using Grid Search Cross Validation had a reasonably significant influence on improving model performance, although the increase was not as substantial as the impact from emotion-based preprocessing. This technique allows for the systematic and measurable determination of the best parameter combination for each algorithm [9].

In the Gradient Boosting model, the combination of a learning_rate of 0.2, a max_depth of 5, and 150 n_estimators was found to improve training stability. The XGBoost model exhibited similar performance with the addition of the colsample_bytree parameter set to 1.0, which permits the utilization of all features during each decision tree construction process. Meanwhile, LightGBM achieved its best results with a configuration of 31 num_leaves and an unrestricted max_depth, allowing the model to learn more complex data structures.

While Grid Search optimization resulted in measurable performance improvements, it is important to consider the associated computational cost. Grid Search inherently increases training time due to the exhaustive evaluation of multiple parameter combinations. However, in this study, the search space was deliberately constrained to a limited set of key hyperparameters and implemented using five-fold cross-validation on a moderately sized dataset. This design balances performance gains with computational feasibility, making the approach practical for applied sentiment analysis tasks. For larger-scale datasets, more efficient optimization strategies, such as Random Search or Bayesian optimization, may be considered to reduce computational overhead while maintaining competitive performance.

Generally, after the application of Grid Search, the accuracy and F1-Score values for all three models were in the 0.91%–0.92% range, signifying that the models had achieved optimal performance for the dataset used. This indicates that the parameter optimization process is capable of reinforcing the results previously obtained from the emotion-based preprocessing stage; thus, the two approaches are mutually complementary in enhancing the quality of sentiment classification. As stated in research [24], grid search hyperparameter tuning has been proven effective in improving model learning efficiency by minimizing the risk of overfitting. However, research [25] emphasized that excessive optimization without proper control via cross-validation can degrade the model's generalization capability.

Although the reported improvements in accuracy and F1-score are relatively modest, the evaluation results consistently show better performance after applying emotion-based preprocessing and Grid Search across all experimental scenarios. This consistency indicates that the observed performance gains are systematic rather than caused by random variation in the dataset. Nevertheless, this study does not claim statistical significance through formal hypothesis testing. Instead, the findings are intended to demonstrate the practical contribution of the proposed approach in improving sentiment classification performance within the experimental scope of this research.

Despite the high accuracy and F1-score achieved after applying Grid Search and emotion-based preprocessing, several misclassification cases were still observed. These errors mainly occurred in comments containing sarcasm, irony, and ambiguous emoji usage, which are prevalent in TikTok interactions. In such cases, comments that were lexically positive but pragmatically negative were often misclassified, as the model primarily relied on explicit lexical features and emotional tokens rather than implicit contextual meaning.

Additionally, polysemous emojis, such as smiling or laughing emojis, contributed to incorrect predictions because their sentiment orientation strongly depended on the surrounding textual context. Although emojis were converted into emotion-related tokens during preprocessing, this transformation was insufficient to fully capture non-literal language patterns. This finding indicates that while emotion-based preprocessing effectively enhances the recognition of explicit emotional expressions, it still faces limitations in handling complex pragmatic and context-dependent sentiment.

The generalizability of the findings in this study is inherently limited by the scope of the dataset, which focuses on a single topic and one public figure. As a result, the performance improvements observed in this research cannot be directly assumed to represent all forms of TikTok discourse or sentiment dynamics across different public figures and discussion topics.

Nevertheless, the proposed emotion-based preprocessing framework is designed to capture general linguistic and emotional characteristics commonly found in TikTok comments, such as informal language, expressive punctuation, and emoji usage. Therefore, while the empirical results are specific to the analyzed case, the methodological approach is potentially applicable to other topics and figures on TikTok, provided that further validation is conducted using more diverse datasets.

Despite the promising results obtained in this study, several limitations should be acknowledged. The sentiment labeling process depends on the quality of the annotation approach, which may introduce bias when handling subjective or context-dependent expressions such as sarcasm or implicit sentiment. In addition, the dynamic nature of informal language and slang on TikTok poses challenges, as linguistic expressions, emoji usage, and emotional cues evolve rapidly over time, potentially reducing the long-term robustness of predefined emotion-based preprocessing rules. Moreover, the dataset was collected from a single topic and public figure, which may lead to sampling bias and limit the representativeness of broader TikTok discourse. Therefore, the findings should be interpreted within the experimental scope of this research.

E. Model Performance Analysis and Implications

The three boosting algorithms utilized—Gradient Boosting, XGBoost, and LightGBM demonstrated distinct characteristics in handling text data. The experimental results show that LightGBM provided the best performance among the three, aligning with its advantages in processing large and complex datasets with high efficiency [23]. XGBoost's performance ranked second, with a performance margin not significantly distant from LightGBM, whereas Gradient Boosting exhibited slightly lower results, which is likely attributable to its less adaptive learning mechanism compared to the other two algorithms [26].

The implications of these results confirm that the combination of emotion-based preprocessing with modern boosting algorithms (particularly LightGBM and XGBoost) constitutes an effective approach for sentiment analysis on Indonesian-language social media. Handling these emotional aspects proved to be critical, given that the linguistic expressions of social media users often do not adhere to formal language structures, such as the use of emojis, capitalization, and expressive punctuation [6]. Furthermore, a boosting-based ensemble approach such as this has proven efficient for text classification due to its ability to capture non-linear relationships among features and enhance model generalization [22]. Therefore, this research contributes to the development of Natural Language Processing (NLP) approaches that are more sensitive to emotional context in unstructured text.

This study focuses on evaluating the impact of preprocessing strategies and hyperparameter optimization within a consistent family of boosting algorithms. While comparisons with other model families such as SVM or transformer-based models are beyond the scope of this work, the controlled experimental setting allows a clearer assessment of how emotion-based preprocessing and Grid Search influence model performance independently of architectural differences.

Beyond its methodological contribution, the findings of this study also offer several practical implications. The proposed sentiment classification framework can be utilized as a public opinion monitoring tool to track changes in public sentiment toward government policies or public figures in near real-time, particularly on highly dynamic platforms such as TikTok.

In a policy analysis context, the sentiment trends identified across different time periods may support policymakers in evaluating public responses to specific actions or announcements. Furthermore, the model can be applied in reputation management, enabling institutions or public figures to detect emerging negative sentiment at an early stage and respond proactively. With further system integration, this approach also has the potential to support real-time sentiment surveillance systems, assisting decision-makers in understanding public reactions during critical events.

F. Analysis of Sentiment Changes Based on Time Period

A sentiment distribution analysis was conducted to examine the variations in public opinion regarding three news contexts concerning the Minister of Finance, Purbaya Yudhi Sadewa, namely: the first week after the handover of office (September 9–15, 2025), an interview after one month in office (October 10, 2025), and the unannounced inspection of secondhand clothing hoarding in Cikarang (November 1, 2025).

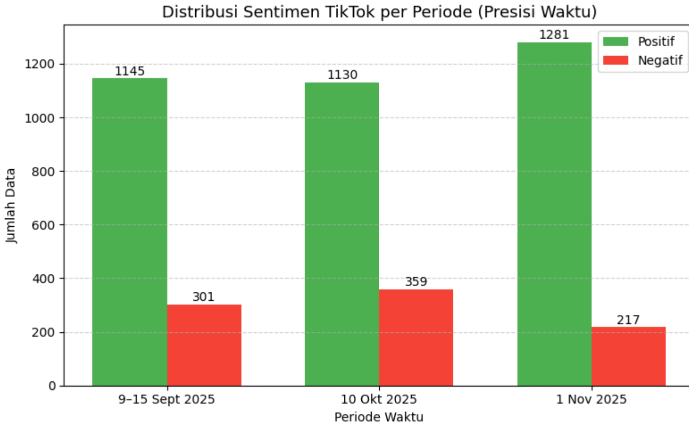

Figure 3. Sentiment Distribution Before Emotion Based

Before the implementation of emotion-based preprocessing, comments during the first period (the initial week of the term of office) were predominantly positive, comprising 1,145 positive comments and 301 negative comments. This indicates public appreciation for the leadership transition that had just taken place. During the one-month-in-office interview period, the number of positive comments slightly decreased to 1,130, whereas negative comments increased to 359. This situation reflects the emergence of public assessment regarding the initial performance and policies implemented. Meanwhile, during the period of the unannounced inspection of hoarded used clothing, positive comments rose to 1,281 and negative comments declined to 217, indicating public support for the decisive action taken.

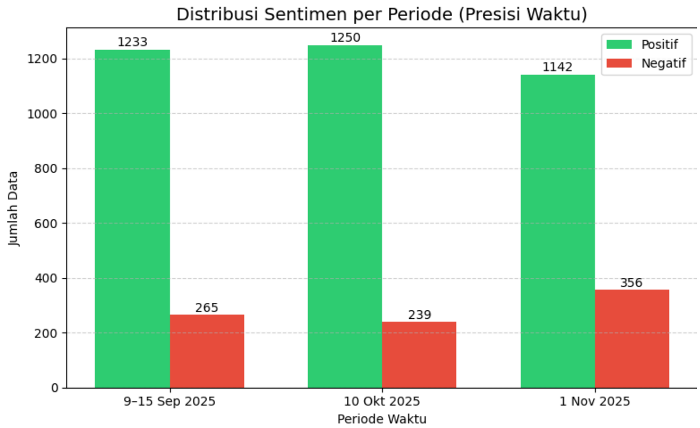

Figure 4. Sentiment Distribution After Emotion-Based

Following the application of emotion-based preprocessing, the sentiment proportions underwent changes in each period. During the first week of the term of office, positive comments increased to 1,233 and negative comments decreased to 265, indicating that this approach was capable of capturing previously undetected expressions of support, such as those conveyed via emojis or capitalization. During the one-month-in-office interview period, positive sentiment rose to 1,250 and negative sentiment declined to 239, signifying that positive emotional expressions, such as optimism and support for initial policies, could be identified more accurately. However, during the Cikarang inspection period, positive comments decreased to 1,142, and negative comments increased to 356. This demonstrates that the emotion-based approach is also more sensitive in recognizing negative emotions, such as anger or sarcasm, implied within the comment text.

Overall, the distributional differences between periods indicate that public opinion on TikTok is contextual and shifts according to developing issues. Each topic elicited distinct emotional reactions, ranging from enthusiasm for the beginning of the leadership term, to evaluations of performance, and responses to policies in the field. This finding aligns with research [1], [2], which confirms that the direction of sentiment and user interaction on TikTok is heavily influenced by the issue's context and public perception of government figures and policies.

IV. CONCLUSION

This study demonstrates that the application of emotion-based preprocessing significantly improves the performance of sentiment classification models on Indonesian-language TikTok comments. Incorporating steps such as capitalization detection, emoji and emoticon conversion, and the handling of expressive punctuation assists the model in recognizing emotional nuances that are frequently overlooked by conventional methods. The best result was achieved by the LightGBM algorithm, attaining an F1-score of 0.92 following hyperparameter optimization using Grid Search, which also enhanced the model's stability and generalization capability.

In addition to performance improvements, the sentiment trend analysis indicates a shift in the proportion of public opinion across time periods. Comments at the beginning of the Minister of Finance Purbaya Yudhi Sadewa's term of office (September 9–15, 2025) were dominated by positive sentiment, with 1,233 positive comments and 265 negative comments. During the first month (October 10, 2025), comments remained predominantly positive, with 1,250 positive comments and 239 negative comments. In the final data collection period (November 1, 2025), sentiment was still observed to be positive, with 1,142 positive and 356 negative comments. This phenomenon demonstrates that public perception on social media is dynamic and readily influenced by the context of evolving issues. Consequently, the combination of emotion-based preprocessing, TF-IDF, SMOTE, and boosting model optimization proved effective not only in enhancing classification accuracy but also in capturing shifts in public emotional tendencies more accurately.

Future research may extend this work by integrating transformer-based models for the Indonesian language, which have demonstrated strong contextual understanding and may better capture non-literal expressions such as sarcasm and irony. Additionally, the use of a richer and dynamically updated emotion lexicon could improve the robustness of emotion-based preprocessing in response to rapidly evolving social media language. Further studies may also explore multimodal sentiment analysis by combining textual data with visual or audio cues from TikTok videos, enabling a more comprehensive understanding of user sentiment and strengthening the applicability of sentiment analysis systems in real-world scenarios.

REFERENCES

[1] Z. Cheng dan Y. Li, "Like, Comment, and Share on TikTok: Exploring the Effect of Sentiment and Second-Person View on the User Engagement with TikTok News Videos," *Soc. Sci. Comput. Rev.*, vol. 42, no. 1, hlm. 201–223, Feb 2024, doi: 10.1177/08944393231178603.

[2] E. Supriyadi dan P. N. Makatita, "Sentiment Analysis of TikTok User Comments on QRIS Adoption in Indonesia Using IndoBERT," *Procedia Comput. Sci.*, vol. 269, hlm. 121–130, Jan 2025, doi: 10.1016/j.procs.2025.08.265.

[3] S. A. A. Hakami, R. Hendley, dan P. Smith, "Emoji Sentiment Roles for Sentiment Analysis: A Case Study in Arabic Texts," dalam *Proceedings of the Seventh Arabic Natural Language Processing Workshop (WANLP)*, H. Bouamor, H. Al-Khalifa, K. Darwish, O. Rambow, F. Bougares, A. Abdelali, N. Tomeh, S. Khalifa, dan W. Zaghouani, Ed., Abu Dhabi, United Arab Emirates (Hybrid): Association for Computational Linguistics, Des 2022, hlm. 346–355. doi: 10.18653/v1/2022.wanlp-1.32.

[4] A. Khan, D. Majumdar, dan B. Mondal, "Sentiment analysis of emoji fused reviews using machine learning and Bert," *Sci. Rep.*, vol. 15, no. 1, hlm. 7538, Mar 2025, doi: 10.1038/s41598-025-92286-0.

[5] M. A. Palomino dan F. Aider, "Evaluating the Effectiveness of Text Pre-Processing in Sentiment Analysis," *Appl. Sci.*, vol. 12, no. 17, hlm. 8765, Jan 2022, doi: 10.3390/app12178765.

[6] H. Tang, W. Tang, D. Zhu, S. Wang, Y. Wang, dan L. Wang, "EMFSA: Emoji-based multifeature fusion sentiment analysis," *PLOS ONE*, vol. 19, no. 9, hlm. e0310715, Sep 2024, doi: 10.1371/journal.pone.0310715.

[7] F.-Y. Chang, "A Quantitative Analysis of Comparison of Emoji Sentiment: Taiwan Mandarin Users and English Users," dalam *Proceedings of the 34th Conference on Computational Linguistics and Speech Processing (ROCLING 2022)*, Y.-C. Chang dan Y.-C. Huang, Ed., Taipei, Taiwan: The Association for Computational Linguistics and Chinese Language Processing (ACLCLP), Nov 2022, hlm. 283–288. Diakses: 9 November 2025. [Daring]. Tersedia pada: https://aclanthology.org/2022.rocling-1.35/

[8] M. A. K. Raiaan *dkk.*, "A systematic review of hyperparameter optimization techniques in Convolutional Neural Networks," *Decis. Anal. J.*, vol. 11, hlm. 100470, Jun 2024, doi: 10.1016/j.dajour.2024.100470.

[9] A. S. Aribowo, N. H. Cahyana, dan Y. Fauziah, "Enhancing Semi-Supervised Sentiment Analysis Through Hyperparameter Tuning Within Iterations: A Comparative Study Using Grid Search and Random Search," dalam *Proceedings of the 2023 1st International Conference on Advanced Informatics and Intelligent Information*

Systems (ICAI3S 2023), vol. 181, A. Putro Suryotomo dan H. Cahya Rustamaji, Ed., dalam Advances in Intelligent Systems Research, vol. 181. , Dordrecht: Atlantis Press International BV, 2024, hlm. 248–260. doi: 10.2991/978-94-6463-366-5_23.

[10] S. W. Iriananda, R. W. Budiawan, A. Y. Rahman, dan I. Istiadi, "Optimasi Klasifikasi Sentimen Komentar Pengguna Game Bergerak Menggunakan Svm, Grid Search Dan Kombinasi N-Gram," *J. Teknol. Inf. Dan Ilmu Komput.*, vol. 11, no. 4, hlm. 743–752, Agu 2024, doi: 10.25126/jtiik.1148244.

[11] S. Matharaarachchi, M. Domaratzki, dan S. Muthukumarana, "Enhancing SMOTE for imbalanced data with abnormal minority instances," *Mach. Learn. Appl.*, vol. 18, hlm. 100597, Des 2024, doi: 10.1016/j.mlwa.2024.100597.

[12] R. Dolak dan P. Kajzar, "Web Scraping and Its Use for Teaching in Course Information Systems in Tourism," dalam *Innovative Technologies and Learning*, W.-S. Wang, F. E. Sandnes, C.-F. Lai, T. A. Sandtrø, dan Y.-M. Huang, Ed., Cham: Springer Nature Switzerland, 2026, hlm. 222–230.

[13] S. Patankar dan M. Phadke, "A CNN-transformer framework for emotion recognition in code-mixed English–Hindi data," *Discov. Artif. Intell.*, vol. 5, no. 1, hlm. 160, Jul 2025, doi: 10.1007/s44163-025-00400-y.

[14] K. S. Eljil, F. Nait-Abdesselam, E. Hamouda, dan M. Hamdi, "Enhancing Sentiment Analysis on Social Media with Novel Preprocessing Techniques," *J. Adv. Inf. Technol.*, vol. 14, no. 6, hlm. 1206–1213, 2023, doi: 10.12720/jait.14.6.1206-1213.

[15] A. Thakkar, D. Mungra, A. Agrawal, dan K. Chaudhari, "Improving the Performance of Sentiment Analysis Using Enhanced Preprocessing Technique and Artificial Neural Network," *IEEE Trans. Affect. Comput.*, vol. 13, no. 4, hlm. 1771–1782, Okt 2022, doi: 10.1109/TAFFC.2022.3206891.

[16] A. R. Lubis, Y. Y. Lase, D. A. Rahman, dan D. Witarsyah, "Improving Spell Checker Performance for Bahasa Indonesia Using Text Preprocessing Techniques with Deep Learning Models," *Ingénierie Systèmes Inf.*, vol. 28, no. 5, hlm. 1335–1342, Okt 2023, doi: 10.18280/isi.280522.

[17] Z. Mansur, N. Omar, S. Tiun, dan E. M. Alshari, "A normalization model for repeated letters in social media hate speech text based on rules and spelling correction," *PloS One*, vol. 19, no. 3, hlm. e0299652, 2024, doi: 10.1371/journal.pone.0299652.

[18] Arif Bijaksana Putra Negara, "The Influence Of Applying Stopword Removal And Smote On Indonesian Sentiment Classification," *Lontar Komput. J. Ilm. Teknol. Inf.*, vol. 14, no. 03, hlm. 172–185, Okt 2025, doi: 10.24843/LKJITI.2023.v14.i03.p05.

[19] K. Machová, M. Mikula, X. Gao, dan M. Mach, "Lexicon-based Sentiment Analysis Using the Particle Swarm Optimization," *Electronics*, vol. 9, no. 8, hlm. 1317, Agu 2020, doi: 10.3390/electronics9081317.

[20] H. Ahmad, W. Akbar, N. Aslam, A. Ahmed, dan M. Khurshid, "TF-IDF Feature Extraction based Sarcasm Detection on Social Media," *J. Comput. Biomed. Inform.*, vol. 5, no. 01, Jun 2023, doi: 10.56979/501/2023.

[21] S. F. Taskiran, B. Turkoglu, E. Kaya, dan T. Asuroglu, "A comprehensive evaluation of oversampling techniques for enhancing text classification performance," *Sci. Rep.*, vol. 15, no. 1, hlm. 21631, Jul 2025, doi: 10.1038/s41598-025-05791-7.

[22] H. Allam, L. Makubvure, B. Gyamfi, K. N. Graham, dan K. Akinwolere, "Text Classification: How Machine Learning Is Revolutionizing Text Categorization," *Information*, vol. 16, no. 2, hlm. 130, Feb 2025, doi: 10.3390/info16020130.

[23] I. Wardhana, Musi Ariawijaya, Vandri Ahmad Isnaini, dan Rahmi Putri Wirman, "Gradient Boosting Machine, Random Forest dan Light GBM untuk Klasifikasi Kacang Kering," *J. RESTI Rekayasa Sist. Dan Teknol. Inf.*, vol. 6, no. 1, hlm. 92–99, Feb 2022, doi: 10.29207/resti.v6i1.3682.

[24] M. Ogunsanya, J. Isichei, dan S. Desai, "Grid search hyperparameter tuning in additive manufacturing processes," *Manuf. Lett.*, vol. 35, hlm. 1031–1042, Agu 2023, doi: 10.1016/j.mfglet.2023.08.056.

[25] I. V. Tetko, R. van Deursen, dan G. Godin, "Be aware of overfitting by hyperparameter optimization!," *J. Cheminformatics*, vol. 16, no. 1, hlm. 139, Des 2024, doi: 10.1186/s13321-024-00934-w.

[26] C. Bentéjac, A. Csörgő, dan G. Martínez-Muñoz, "A comparative analysis of gradient boosting algorithms," *Artif. Intell. Rev.*, vol. 54, no. 3, hlm. 1937–1967, Mar 2021, doi: 10.1007/s10462-020-09896-5.