TIN: Terapan Informatika Nusantara
Vol 6, No 3, August 2025, page 345-356
ISSN 2722-7987 (Media Online)
Website https://ejurnal.seminar-id.com/index.php/tin
DOI 10.47065/tin.v6i3.8210

Sentiment Analysis of Public Opinion on Facebook Monetization in Social Media Using the SVM Algorithm

Nurmaiyah, Aidil Halim Lubis*

Department of Computer Science, State Islamic University of North Sumatra, Indonesia
Email: [1]nurmaiyahsipahutar@gmail.com, [2,*]aidilhalimlubis@uinsu.ac.id
Correspondence Author Email: aidilhalimlubis@uinsu.ac.id

Abstract–Sentiment analysis on Facebook's monetization policy has become a significant topic in the era of rapid digital transformation. This study examines public opinion on the policy by analyzing TikTok user comments that specifically discuss Facebook monetization. TikTok was chosen as the data source because it reflects spontaneous and real-time public reactions, including discussions about other platform policies. A total of 5,000 TikTok comments were collected using web scraping techniques. The data underwent several preprocessing stages, including text cleaning, tokenization, normalization, stopword removal, and stemming. Sentiment labeling was carried out using the Indonesian Sentiment Lexicon (InSet), while feature extraction employed the Term Frequency–Inverse Document Frequency (TF-IDF) method. The classification process was conducted using the Support Vector Machine (SVM) algorithm with a linear kernel. The dataset was split into training and testing sets with an 80:20 ratio. The classification achieved an accuracy of 80%, with a precision of 80% for both positive and negative sentiments, recall scores of 81% and 79%, and F1-scores of 81% and 79%, respectively. These findings demonstrate that integrating TF-IDF weighting with the SVM algorithm is effective for automatically classifying public sentiment toward social media monetization policies. Furthermore, this study provides insights into public reactions to Facebook monetization from the perspective of TikTok users, thereby contributing to an understanding of how monetization policies influence user sentiment on social media platforms.

Keywords: Support Vector Machine (SVM); Sentiment Analysis; Facebook Monetization; TF-IDF; TikTok Comments

1. INTRODUCTION

Rapid development of information and communication technology has encouraged the public to become increasingly active on social media as a means of expressing opinions and sharing information (Tei et al., 2025). One notable innovation arising from this trend is the implementation of content monetization policies by platforms such as Facebook (Iosifidis, 2025). Through the Facebook Professional feature, users can generate income from their uploaded content, creating new economic opportunities while also provoking diverse public reactions, including support, privacy concerns, and debates over content quality and ethics (Aguiar et al., 2022).

Meanwhile, TikTok has emerged as a leading platform where users express opinions and respond to current issues in real time (Moreno et al., 2021). Users are often spontaneous and candid, making them an authentic data source for capturing public sentiment. As such, TikTok functions not only as entertainment media but also as a dynamic arena for public discourse (Grantham et al., 2025) (Tangke et al., 2024).

Sentiment analysis, an application of natural language processing that classifies public opinion based on textual expression patterns, has proven effective for understanding user perceptions (Srinidhi et al., 2024). Among various classification techniques, the Support Vector Machine (SVM) algorithm remains popular due to its strong performance in text classification tasks (Lisa, 2023).

Previous studies have demonstrated the success of SVM in public-opinion analysis. For example, SVM achieved high accuracy in classifying sentiments toward PSBB regulations into positive, negative, and neutral categories (Fadhila Sari & Salsabila, 2024). In another study, SVM outperformed Naïve Bayes with an 84% accuracy rate when classifying TikTok user reviews (Indriyani et al., 2023). and it achieved 73% accuracy in analyzing sentiment toward the 2024 KPU (Maulana & Putri, 2024). While some works have examined Facebook monetization from Islamic legal and ethical perspectives (Mohamed et al., 2023), quantitative sentiment-analysis of public opinion on this policy remains limited.

Recent studies in social media sentiment analysis increasingly use hybrid methods combining lexicon-based labeling, TF-IDF, and machine learning, achieving better accuracy. Additionally, short-video platforms like TikTok are considered real-time data sources driven by youth, although research on Facebook monetization in this context remains limited. This study integrates these approaches to fill a critical gap in understanding public sentiment toward Facebook's monetization policy.

To date, few studies have applied text-classification methods to quantify reactions to Facebook's monetization policy, and no research has leveraged the youth-driven, real-time TikTok comment environment for this purpose. This study addresses these gaps by collecting 5,000 TikTok comments related to Facebook monetization and implementing a hybrid approach—lexicon-based labeling, TF-IDF weighting, and SVM classification—to produce a nuanced, data-driven understanding of public sentiment.

The main contributions of this research include the comprehensive identification and quantification of public sentiment toward Facebook's content monetization policy through the analysis of real-time TikTok comments, the development of a robust hybrid sentiment-analysis pipeline that integrates lexicon-based labeling, TF-IDF feature extraction, and Support Vector Machine classification, and an empirical evaluation that demonstrates the proposed model's high accuracy and the generation of actionable insights for platform decision makers. By clarifying the specific

TIN: Terapan Informatika Nusantara
Vol 6, No 3, August 2025, page 345-356
ISSN 2722-7987 (Media Online)
Website https://ejurnal.seminar-id.com/index.php/tin
DOI 10.47065/tin.v6i3.8210

gap in existing literature and outlining these principal contributions, this Introduction establishes a foundation for the subsequent methodology and results sections.

2. RESEARCH METHODOLOGY

This study began with a planning stage, namely the determination of the topic about public sentiment analysis toward Facebook's monetization policy, based on user comments on the TikTok platform, using the Support Vector Machine (SVM) algorithm. Monetization itself is a process of converting digital assets such as content, services, or social media platforms into income sources. In this context, Facebook applies monetization through various features and policies that affect the user experience, which in turn causes diverse public responses (Shahbazi et al., 2025). The approach used in this study is quantitative in nature, with a focus on text data analysis from social media platforms.

Sentiment analysis is a method of processing that connects the computing of opinions, feelings, or responses from individuals or groups toward certain events expressed in the form of text. This method is capable of transforming unstructured data into structured data and can be applied to various needs, such as evaluating businesses, programs, products, applications, and so on (Kah & Zeroual, 2022).

One of the core techniques in this study is the Support Vector Machine algorithm, which is widely known for its accuracy and performance in classification tasks. SVM works by separating two classes of data using an optimal hyperplane and maximum margin. In the context of sentiment analysis, textual data is first converted into vector representations using the TF-IDF method before being processed by the SVM model. However, SVM may face limitations when dealing with data that has non-linear patterns or high similarity between classes. Therefore, to improve model performance, kernel functions such as linear, RBF, and polynomial are used to map data into higher-dimensional spaces. The selection of the appropriate kernel and tuning of model parameters becomes a crucial factor in determining model accuracy (Rodríguez-Ibánez et al., 2023).

Before the classification process is conducted, a preprocessing stage is required to ensure that the text data used is clean, structured, and aligned with the required format (Hayadi & Maulita, 2025). The initial step in preprocessing is cleaning, which involves removing all non-alphabetic characters from the text so that only letters remain. Next, case folding is performed to convert uppercase letters to lowercase, which is important for resolving differences in word forms. The tokenization process then breaks the text into smaller units such as words or phrases to simplify further analysis. This study utilizes the JMESPath query language and library for extracting and transforming JSON data. A total of 5,000 TikTok comments related to Facebook monetization were collected via web scraping using an unofficial TikTok API endpoint, with the results stored in JSON format. This format preserves hierarchical metadata, such as username, comment text, timestamp, and engagement metrics. JMESPath enables efficient filtering and transformation of this structured JSON data, ensuring that only relevant textual fields are retained for sentiment analysis while unnecessary metadata is discarded (Aufar et al., 2023).

The next stage is normalization, which involves converting informal word forms such as abbreviations, emoticons, or slang into standardized forms that can be recognized by the system. After normalization, stopword removal is carried out to eliminate common words that do not hold significant meaning for the classification process, such as conjunctions or articles. Finally, stemming is performed to strip affixes from words and return them to their root forms, thereby reducing variation in words with the same meaning (Mulla et al., 2025).

Once the text is preprocessed, the TF-IDF method is applied to assign weights to each word in the document based on its frequency and uniqueness in the corpus. This technique is known for being efficient, easy to use, and capable of assigning relevant word weights for information analysis. The results of this weighting process serve as input for the classification process using SVM (Didi et al., 2022) (Maia et al., 2025).

This study utilizes the Python programming language (version 3.8 or later) on the Google Colaboratory platform to implement a sentiment analysis model. Python 3.8+ offers enhanced stability and performance, with features such as assignment expressions (PEP 572) and improved garbage collection that support efficient text preprocessing and model training. Its compatibility with scikit-learn version 1.x ensures seamless integration for implementing TF-IDF and Support Vector Machine (SVM) algorithms (Saleh et al., 2022). Scikit-learn provides robust functionality, including optimized matrix operations and built-in support for pipelines, making it highly suitable for combining preprocessing stages such as tokenization, normalization, and vectorization with classification tasks. Google Colaboratory (Colab) serves as an accessible cloud-based environment that supports Python 3.8+ and scikit-learn 1.x out of the box. It enables researchers to run experiments using free CPU, GPU, or TPU resources without local configuration. Colab's integration with Google Drive allows for efficient data storage and management, while real-time collaboration features facilitate joint development and review. Additionally, its support for popular visualization libraries enhances the monitoring and interpretation of preprocessing results and model performance (Wilyani et al., 2024).

For evaluate Classification model performance, confusion matrix is used. Table this compare results model predictions with actual data, as well as describe quantity and type errors that occur. Prediction data is value obtained from results machine learning modeling, while the actual data is mark actually owned (Markoulidakis & Markoulidakis, 2024). From the confusion matrix, it can be counted metric evaluation important such as accuracy, precision, recall, and F1-score, which provide description comprehensive about effectiveness of the model in classify data accurate (Obi, 2023).

TIN: Terapan Informatika Nusantara
Vol 6, No 3, August 2025, page 345-356
ISSN 2722-7987 (Media Online)
Website https://ejurnal.seminar-id.com/index.php/tin
DOI 10.47065/tin.v6i3.8210

Figure 1 presents the overall framework of this study, illustrating the sequential steps from data collection to classification

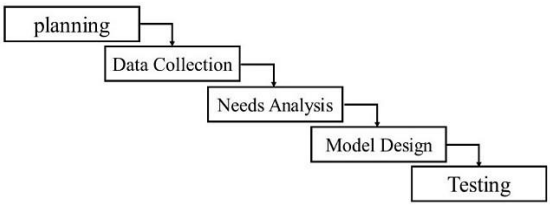

Figure 1. Framework of the study

Figure 2 illustrates the model design flowchart, which integrates preprocessing, feature extraction, and SVM classification.

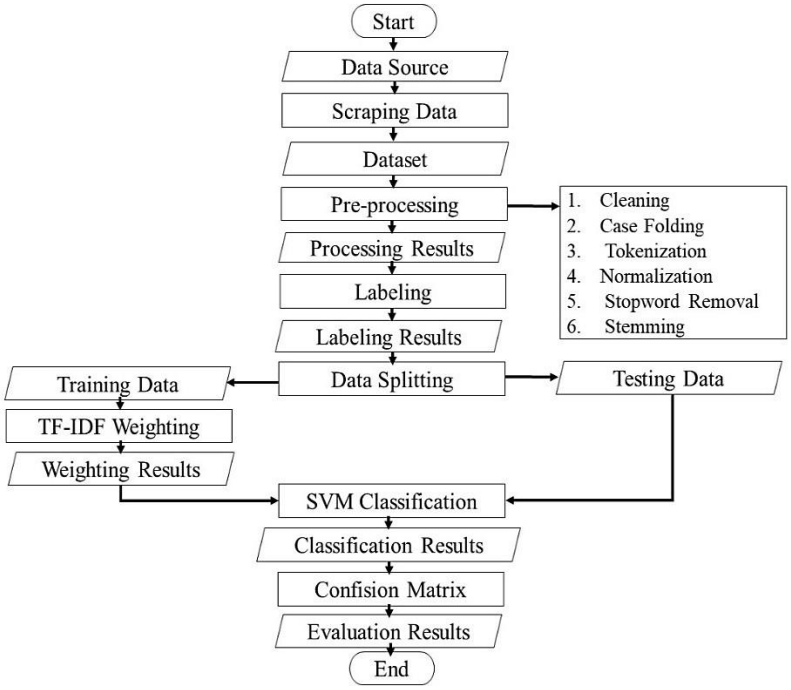

Figure 2. Model design flowchart

3. RESULTS AND DISCUSSION

3.1 Data Source

Dataset in study this obtained through the data scraping process against comments on some TikTok videos popular topics that discuss Facebook monetization, namely from account neliagustinmakeup, mydanas, rickyerlangga05, triokamvret, and ginicaranyaofficial. Total data collected as many as 5,000 comments, with the scraping process carried out on 9 February 2025.

3.2 Data scraping

In study this, scraping is done via TikTok API endpoints that are not official, with help Python language and JMESPath query/library for extraction JSON elements. This process executed using Google Colab. Figure 3 shows the structure of the data obtained from the scraping process.

	Username	Nickname	Komentar	Tanggal	Like Count	Comment ID
0	produksunda AE	dian35804	mau ikutan fb pro tpi hidup ku privasi	21 January 2024	497	7326554099259097862
1	YasnaSudarmanto	maulidayasna06	apa cuma q yg gak paham fb pro	22 January 2024	187	7326774032416736006
2	V-Rima	tigaal123	semenjak ada fb pro jd tau isi rmh orng2, dan menu mknnan mereka	23 January 2024	226	7327230274131378950
3	Louisianna yoelanda w.Foess	yoelanda_wf	aku punya dari 2008 dan aku close permanen 2020...udh gak tau bunda2 ala2 FB pro	21 January 2024	55	7326377052579037958
4	YAMEROOOOOO	marga_a14	fb sekarang milik Bunda bunda	22 January 2024	246	7326773683690685190

Figure 3. Sample data scraping on comments tiktok

TIN: Terapan Informatika Nusantara
Vol 6, No 3, August 2025, page 345-356
ISSN 2722-7987 (Media Online)
Website https://ejurnal.seminar-id.com/index.php/tin
DOI 10.47065/tin.v6i3.8210

Referring to Figure 3 above, the data scraping results contain six columns detailing the comments, which include: Username, Nickname, Comment, Comment ID, Date, and Like Count. The researcher removed duplicate comments, such as entries with identical text or repetitive emojis, reducing the dataset from 5,000 to 3,264 entries. In the subsequent preprocessing stage, the data was further cleaned by removing blank entries, including comments with no text or those containing only spaces, using the command df = data.dropna(). This process resulted in a final dataset of 3,000 entries. Although the dataset size decreased, this step enhanced the quality of the analysis by ensuring that only informative and relevant comments were retained for feature extraction, allowing the model to learn more effectively.

3.3 Preprocessing process

The preprocessing stage involves transforming the dataset into a structured and usable format that can be processed by the system in the next stage. This process includes cleaning, case folding, tokenization, normalization, stopword removal, and stemming. Table 1 presents a comparison between raw text and its processed form after text cleaning, case folding, tokenization, normalization, stopword removal, and stemming. This step is essential for preparing the data for machine learning.

Table 1. Example of raw review and its preprocessed result

Raw Text	After Pre-processing
FB pro but still don't want to make a day in my life...just posts about traveling and quotes, still trying to be aesthetic □	fb pro doesn't want to make a day my life post with aesthetic effort quotes.
How do I make a pro Facebook account? Want to join in? □	how to make fb pro join
It's so full of mothers all the time □ Morning hello mother afternoon hello mother evening hello mother until I'm too lazy to scroll through Facebook □□□□	very full of content, hello morning, hello afternoon, hello evening, hello lazy mother, scrolling through Facebook
FB Pro is a headache, even though I'm diligent about uploading. Greetings from the interaction.	fb pro makes you dizzy, diligently uploading greetings, interactions
Bismillah this year I got a salary from fb pro □□	Bismillah, fb pro salary

3.4 Labeling

This study adopts a lexicon-based approach using the Indonesian Sentiment Lexicon (InSet), developed by Koto and Rahmaningtyas (2017), which contains over 3,069 positive words and 6,609 negative words, each assigned a sentiment weight ranging from –5 to +5. Table 2 presents sample results from the labeling process.

Table 2. Sample sentiment labeling results

Text preprocessing results	Word weight identification	Total weight	Sentiment labels
tired of making content, it's okay, fb pro	[4, -4, 0, 0, -5, 0, 0, 0]	-5	Negative
please help support me, mom, hahaha	[3, -4, 4, 0, 3, -5, 3]	4	Positive

The labeling process using the InSet lexicon categorized the comments into two classes: positive and negative. Of the total 3,000 comments analyzed, 1,520 comments (50.67%) were classified as negative, while 1,480 comments (49.33%) were categorized as positive, as shown in Figure 4. This nearly balanced distribution indicates that public opinion on Facebook's monetization policy is divided: some users are enthusiastic about the potential income opportunities offered, whereas others express dissatisfaction due to perceived limitations or inequities in the program. Such an even split provides insight into the diverse reactions among TikTok users, which is important for platform managers to consider when designing strategies for user retention and content diversity.

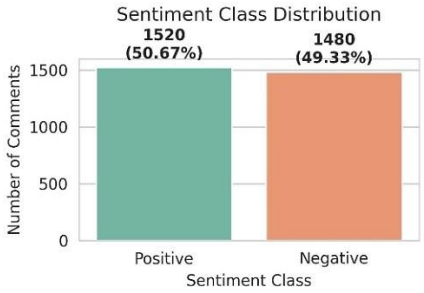

Figure 4. Percentage and frequency of sentiment classes

3.5 Data Splitting

Data splitting is the process of dividing a dataset into two main parts to effectively evaluate the performance of a machine learning model. This step is essential to ensure that the model can generalize well to unseen data by training it on one subset and validating it on another. At this stage, the researcher divides the dataset into 80% training data and 20% testing data. This proportional division is shown in Figure 5.

TIN: Terapan Informatika Nusantara
Vol 6, No 3, August 2025, page 345-356
ISSN 2722-7987 (Media Online)
Website https://ejurnal.seminar-id.com/index.php/tin
DOI 10.47065/tin.v6i3.8210

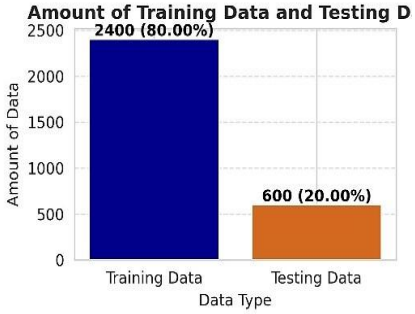

Figure 5. Proportional division of dataset into training and testing subsets

3.6 TF-IDF weighting

After the sentiment labeling and text cleaning process, word weighting was performed using the Term Frequency–Inverse Document Frequency (TF-IDF) method. This method calculates the frequency of word occurrences in a document (TF) and multiplies it by the inverse document frequency (IDF) to emphasize words that are unique within the data corpus.

Table 3 presents examples of the training data that have been tokenized into terms along with their corresponding sentiment classes. Table 4 displays test data samples prepared for evaluation. Meanwhile, Table 5 shows the document frequency (DF) values of terms in the training dataset, which are used in the IDF calculation.

Table 3. Sample of training data

Training Sentiment	Class
["tutorial", "content", "fb", "pro"]	Positive
["greetings", "interaction", "mother"]	Positive
["way", "make", "fb", "pro"]	Positive
["retreat", "tired", "interaction", "comment", "enthusiasm"]	Negative
["monetization", "fb", "people", "make", "content", "let", "money", "don't", "care", "good"]	Negative

Table 4. Sample of test data

Test Sentiment
["spirit ", " regards ", " know", "mutual", "support", "yes", "mother"]
["fb", "pro", "make", "person", "thirsty", "interaction", "content", "zero"]

Table 5. Document frequency (DF) values of training data

Term	TF					DF
	D1	D2	D3	D4	D5	
tutorial	1	0	0	0	0	1
content	1	0	0	0	1	2
fb	1	0	1	0	1	3
…	…	…	…	…	…	…
good	0	0	0	0	1	1

The inverse document frequency (IDF) for each word is then calculated using the formula shown in Equation (1).

$$IDF = In\left(\frac{D+1}{df+1}\right) + 1 \quad (1)$$

As illustration, below implementation formula on the first data:

$$IDF_{tutorial} = \text{In}\left(\frac{D+1}{df+1}\right) + 1 = \text{In}\left(\frac{5+1}{1+1}\right) + 1 = \text{In}\,(3) + 1 = 2.099.$$

As shown in Table 6, the IDF values vary Based on the frequency of each term.

Table 6. Inverse document frequency (IDF) values of training data

Term	DF	IDF
tutorial	1	2,099
content	2	1,693
fb	3	1,405
…	…	…

TIN: Terapan Informatika Nusantara
Vol 6, No 3, August 2025, page 345-356
ISSN 2722-7987 (Media Online)
Website https://ejurnal.seminar-id.com/index.php/tin
DOI 10.47065/tin.v6i3.8210

Term	DF	IDF
good	1	2,099

TF-IDF values are calculated after TF and IDF values are known, accordingly with Equation (2).

$$W = TF \times IDF \quad (2)$$

As illustration, below implementation formula on the first data:

$$W = TF \times IDF = 1 \times 2.099 = 2.099.$$

As shown in Table 7, the TF-IDF values were computed based on the training data.

Table 7. TF-IDF values of training data

Term	TF-IDF				
	D1	D2	D3	D4	D5
tutorial	2,099	0	0	0	0
content	1,693	0	0	0	1,693
fb	1,405	0	1,405	0	1,405
…	…	…	…	…	…
good	0	0	0	0	2,099

The TF-IDF values are normalized using Equation (3), which ensures that all values fall within a standardized range.

$$TF_{norm}(t,d) = \frac{TF(t,d)}{\sqrt{\sum_i (TF(t,d))^2}} \quad (3)$$

As illustration, below implementation formula on the first data:

$$TF_{norm}(t,d) = \frac{TF(t,d)}{\sqrt{\sum_i (TF(t,d))^2}} = \frac{2.099}{\sqrt{(2.099)^2 + (1.693)^2 + (1.405)^2 + (1.693)^2}} = \frac{2.099}{\sqrt{12.1122}} = \frac{2.099}{3.4802} = 0.603.$$

Table 8 presents the results of the data normalization process.

Table 8. Normalized TF-IDF values

No	Term	D1	D2	D3	D4	D5
1	tutorial	0.603	0	0	0	0
2	content	0.486	0	0	0	0.273
3	fb	0.404	0	0.404	0	0.226
…	…	…	…	…	…	…
20	good	0	0	0	0	0.338

3.7 Support Vector Machine Classification

After the data was cleaned and arranged, the classification process was carried out using the Support Vector Machine (SVM) algorithm. The dataset was divided in an 80:20 ratio for training and testing. The training data was used to learn the pattern of each class, while the testing data was used to evaluate the classification accuracy.

3.7.1 Linear Kernel

During the classification process, a linear kernel was used because the analyzed data demonstrated a linear pattern. Equation (4) represents the linear kernel function used to compute the similarity between data points in the SVM model.

$$K(x,y) = x \times y \quad (4)$$

The following is an example of data representation for 5 data items explained in Table 9

Table 9. Normalized data representation

	$x1$	$x2$	$x3$	$x4$	$x5$
$y1$	$K(x1, y1)$	$K(x2, y1)$	$K(x3, y1)$	$K(x4, y1)$	$K(x5, y1)$
$y2$	$K(x1, y2)$	$K(x2, y2)$	$K(x3, y2)$	$K(x4, y2)$	$K(x5, y2)$
$y3$	$K(x1, y3)$	$K(x2, y3)$	$K(x3, y3)$	$K(x4, y3)$	$K(x5, y3)$
$y4$	$K(x1, y4)$	$K(x2, y4)$	$K(x3, y4)$	$K(x4, y4)$	$K(x5, y4)$
$y5$	$K(x1, y5)$	$K(x2, y5)$	$K(x3, y5)$	$K(x4, y5)$	$K(x5, y5)$

The linear kernel function was applied to compute the similarity between data points. For instance, in the first row of the calculation, the resulting kernel value was 0.364. A summary of the computed kernel values is presented in Table 10.

TIN: Terapan Informatika Nusantara
Vol 6, No 3, August 2025, page 345-356
ISSN 2722-7987 (Media Online)
Website https://ejurnal.seminar-id.com/index.php/tin
DOI 10.47065/tin.v6i3.8210

Table 10. Linear kernel calculation results

No	1	2	3	4	…	17	18	19	20
1	0.364	0.293	0.244	0.293	...	0	0	0	0
2	0.293	0.364	0.258	0.236	...	0.092	0.092	0.092	0.092
3	0.244	0.364	0.378	0.393	...	0.076	0.076	0.076	0.076
…	...	...	...	...	...	...	...	...	...
20	0	0.092	0.076	0	...	0.114	0.114	0.114	0.114

Once the kernel value is obtained, the next step is next is count Hessian matrix. Before the calculation process done, determined moreover formerly several parameters such as αi, C, γ, λ, and the number iteration maximum. Explanation regarding the parameters used in calculation the Hessian matrix is presented in Table 11.

Table 11. Parameter values used in training

Parameter	Mark
αi	0
C	1
γ	0.1
λ	0.5
iteration	2

3.7.2 Calculation Hessian Matrix

The Hessian matrix is calculated using the kernel values between data points, along with their respective class labels, as shown in Equation (5).

$$D_{ij} = y_i y_j (K(x_i x_j) + \lambda^2 \quad (5)$$

Example calculation mark Hessian matrix in column 1 row 1:

$$D_{11} = y_i y_j (K(x_i x_j) + \lambda^2 = 1 \times 1(0.364) + 0.5^2 = 0.614.$$

The computed Hessian matrix is shown in Table 12. This matrix quantifies the interaction between data points using the kernel function, reflecting both self-similarity (on the diagonal) and pairwise similarity (off-diagonal). For readability, only a portion of the matrix is displayed.

Table 12. Hessian matrix calculation results

No	1	2	3	4	…	17	18	19	20
1	0.614	0.543	0.494	0.543	…	0.25	-0.25	0.25	0.25
2	0.543	0.561	0.508	0.486	…	0.342	-0.342	0.342	0.342
3	0.494	0.508	0.628	0.643	…	0.326	-0.326	0.326	0.326
…	…	…	…	…	…	…	…	…	…
20	0.25	0.342	0.326	0.25	…	0.364	-0.364	0.364	0.364

3.7.3 Sequential Training SVM

The alpha values are updated sequentially using the following rule in Equation (6), which considers the learning rate and error term.

$$E_i \sum_{j=1}^{n} a_j D_{ij} \quad (6)$$

In the calculation initial, iteration started from iteration 0. Because the value α beginning still has a value of 0. Example calculation in column 1 row 1:

$$E_i \quad = \sum_{j=1}^{n} a_j D_{ij} \ = 0 \times 0.614 = 0.$$

Table 13 presents the error values Ei in the initial iteration (iteration 0), assuming all alpha values start from zero. These values are used in the update process for the SVM training.

Table 13. Initial iteration values (iteration 0)

No	E_i
1	0
2	0
3	0
…	…
20	0

TIN: Terapan Informatika Nusantara
Vol 6, No 3, August 2025, page 345-356
ISSN 2722-7987 (Media Online)
Website https://ejurnal.seminar-id.com/index.php/tin
DOI 10.47065/tin.v6i3.8210

Next, it is done calculation of δα in iteration first for determine value of α. The equation for get δα at iteration 0. Example calculation on the first data. $\delta\alpha_0 = \min\{\max[\gamma(1 - E_i, -a_i], C - a_i\} = \min\{\max[0.1\ (1 - 0), -0], 1 - 0\} = \min\{\ 0.1{,}1\ \} = 0.1$. In iteration 0, the change in each alpha (Δα) is calculated using the update rule. Table 14 shows the resulting delta values, which are then used to update alpha in the next step.

Table 14. Delta α values from iteration 0

No	Δα
1	0.1
2	0.1
3	0.1
…	…
20	0.1

Once Δα is obtained, the alpha value at iteration t + 1 is updated using Equation (7).

$$\alpha_{baru} = \alpha_i + \delta\alpha_i \quad (7)$$

Example calculation for the first data.

$\alpha_{baru} = \alpha_i + \delta\alpha_i = 0 + 0.1 = 0.1$

Based on the delta values from the previous step, the updated alpha (αi) values are presented in Table 15. These values will continue to be refined through subsequent iterations

Table 15. Updated α values in iteration 1

No	α
1	0.1
2	0.1
3	0.1
…	…
20	0.1

The calculation is performed sequentially up to the maximum iteration limit of two. This results in two iterations used to determine the α_i values for identifying support vectors. The final alpha values are shown in Table 16, where non-zero values indicate the data points that serve as support vectors defining the decision boundary.

Table 16. Final α values after iteration 2

No	α
1	0.15306
2	0.15051
3	0.14497
…	…
20	0.1567

After calculation Previously, the support vector was determined based on latest α value with take mark highest from each class, as listed in Table 17.

Table 17. Support vector determination

No	D1	D2	D3	D4	D5	α	Class
1	0.603	0	0	0	0	0.15306	1
2	0.486	0	0	0	0.273	0.15051	1
3	0.404	0	0.404	0	0.226	0.14497	1
…	…	…	…	…	…	…	…
20	0	0	0	0	0.338	0.1567	1

Next step is count kernel function of each class with use the α value of Hessian matrix of the 8th and 9th columns. Manual calculation for the K (x_i x^+) of the positive class is 0.15306, while the value K (x_i x^-) of class negative is 0.15051.

$$K(x_i x^+ = \sum a_i y_i D_i) = (0.15306 \times 1 \times 0.25) + (0.15051 \times 1 \times 0.325) + (0.14497 \times 1 \times 0.508) + (0.14577 \times 1 \times 0.486) + (0.15442 \times 1 \times 0.25) + (0.15507 \times 1 \times 0.25) + (0.15442 \times 1 \times 0.25) + (0.15306 \times 1 \times 0.543) + (0.15051 \times 1 \times 0.561) + (0.23674 \times -1 \times -0.25) + (0.23674 \times -1 \times -0.25) + (0.16326 \times 1 \times 0.25) + (0.16326 \times 1 \times 0.25) + (0.1567 \times 1 \times 0.25) + (0.1567 \times 1 \times 0.25) + (0.1567 \times 1 \times 0.342) + (0.1567 \times 1 \times 0.342) + (0.2433 \times -1 \times -0.342) + (0.1567 \times 1 \times 0.342) + (0.1567 \times 1 \times 0.342) = 1.091119$$

TIN: Terapan Informatika Nusantara
Vol 6, No 3, August 2025, page 345-356
ISSN 2722-7987 (Media Online)
Website https://ejurnal.seminar-id.com/index.php/tin
DOI 10.47065/tin.v6i3.8210

$$K(x_i x^- = \sum a_i y_i D_i) = (0.15306 \times 1 \times -0.25) + (0.15051 \times 1 \times -0.25) + (0.14497 \times 1 \times -0.25) + (0.14577 \times 1 \times -0.25) + (0.15442 \times 1 \times -0.25) + (0.15507 \times 1 \times -0.424) + (0.15442 \times 1 \times -0.25) + (0.15306 \times 1 \times -0.25) + (0.15051 \times 1 \times -0.25) + (0.23674 \times -1 \times 0.465) + (0.23674 \times -1 \times 0.465) + (0.16326 \times 1 \times -0.465) + (0.16326 \times 1 \times -0.465) + (0.1567 \times 1 \times -0.25) + (0.1567 \times 1 \times -0.25) + (0.1567 \times 1 \times -0.25) + (0.1567 \times 1 \times -0.25) + (0.2433 \times -1 \times 0.25) + (0.1567 \times 1 \times -0.25) + (0.1567 \times 1 \times -0.25) = -1.035305$$

Table 18 presents the calculated values of x^+ and x^- used for final classification.

Table 18. Computed values of x+ and x-

x^+	x^-
1.091119	-1.035305

Once the support vectors are identified, the bias term is calculated using Equation (8).

$$b = \frac{1}{2}[\sum_{i=1} a_i y_i K(x_i x^+) + \sum_{i=1} a_i y_i K(x_i x^-)] \quad (8)$$

From the equation above, then obtained bias value as following:

$$b = -\frac{1}{2}[w \times x^+ + w \times x^-] = -\frac{1}{2}[1.091119 + (-1.035305)] = -\frac{1}{2}(0.055814) = -0.027907.$$

Stage testing started after bias value is obtained, with step beginning count TF-IDF values for all test data is explained like Table 19.

Table 19. TF-IDF values for test data

No	Term	TF	DF	IDF	TF-IDF	Normalization
1	spirit	1	1	1.405	1.405	0.378
2	regards	1	1	1.405	1.405	0.378
3	know	1	1	1.405	1.405	0.378
…	…	…	…	…	…	…
20	zero	1	1	1.405	1.405	0.3536

The next step is to calculate the kernel values between each test data and the training data. The results of the kernel calculations are shown in Tables 20 and 21. The following is the calculation for column 1, row 1: $K(x, y) = (x1 \times y1) + (x1 \times y2) + . + (x1 \times y5) = (0.378 \times 0.206) + (0.378 \times 0) + (0.378 \times 0) + (0.378 \times 0) + (0.378 \times 0) = 0.078$

Table 20. Kernel values between training data and test data d1

No	spirit	regards	know	mutual	support	yes	mother
1	0.078	0.078	0.078	0.078	0.078	0.078	0.078
2	0.126	0.126	0.126	0.126	0.126	0.126	0.126
3	0.078	0.078	0.078	0.078	0.078	0.078	0.078
…	…	…	…	…	…	…	…
20	0.078	0.078	0.078	0.078	0.078	0.078	0.078

Table 21. Kernel values between training data and test data d2

No	fb	pro	make	person	thirsty	interaction	content	zero
1	0.073	0.073	0.073	0.073	0.073	0.073	0.073	0.073
2	0.118	0.118	0.118	0.118	0.118	0.118	0.118	0.118
3	0.073	0.073	0.073	0.073	0.073	0.073	0.073	0.073
…	…	…	…	…	…	…	…	…
20	0.073	0.073	0.073	0.073	0.073	0.073	0.073	0.073

The kernel value that has been obtained used for count test data weight according to with Equality the following, for example in row 1 column 1. Tables 22 and 23 show the weighted term values for test data 1 and test data 2, respectively.

$$w \times x = a_i y_i K(x_i x_j) = 0.15306 \times 1 \times 0.078 = 0.012$$

Table 22. Term weighting on test data d1

No	spirit	regards	know	mutual	support	yes	mother
1	0.012	0.012	0.012	0.012	0.012	0.012	0.012
2	0.019	0.019	0.019	0.019	0.019	0.019	0.019

TIN: Terapan Informatika Nusantara
Vol 6, No 3, August 2025, page 345-356
ISSN 2722-7987 (Media Online)
Website https://ejurnal.seminar-id.com/index.php/tin
DOI 10.47065/tin.v6i3.8210

No	spirit	regards	know	mutual	support	yes	mother
3	0.011	0.011	0.011	0.011	0.011	0.011	0.011
…	…	…	…	…	…	…	…
20	0.012	0.012	0.012	0.012	0.012	0.012	0.012
Σ	0.176	0.176	0.176	0.176	0.176	0.176	0.176

Table 23. Term weighting on test data d2

No	fb	Pro	make	person	thirsty	interaction	content	zero
1	0.011	0.011	0.011	0.011	0.011	0.011	0.011	0.011
2	0.018	0.018	0.018	0.018	0.018	0.018	0.018	0.018
3	0.011	0.011	0.011	0.011	0.011	0.011	0.011	0.011
…	…	…	…	…	…	…	…	…
20	0.011	0.011	0.011	0.011	0.011	0.011	0.011	0.011
Σ	0.163	0.163	0.163	0.163	0.163	0.163	0.163	0.163

Then mark function f(x) counted from amount test data weight. Based on the hyperplane, the data is classified as positive If w × x + b = 1 and negative If w × x + b = -1. Positive result show class positive, whereas results negative show class negative.

Test Data 1

["spirit ", " regards ", " know", "mutual", "support", "yes", "mother"]

$$f(x) = w \times x + b = \Sigma a_i y_i K(x_i x_j) + b = (0.176 + 0.176 + 0.176 + 0.176 + 0.176 + 0.176 + 0.176 + (-0.027907) = 1.204$$

The classification function yields sign(1.204) = 1, indicating a positive sentiment.

Test Data 2

["fb", "pro", "make", "person", "thirsty", "interaction", "content", "zero"]

$$f(x) = w \times x + b = \Sigma a_i y_i K(x_i x_j) + b = (0.163 + 0.163 + 0.163 + 0.163 + 0.163 + 0.163 + 0.163 + 0.163 + (-0.027907) = 1.276$$

The classification function yields sign(1.276) = 1, indicating a positive sentiment.

After carrying out the testing process on 2 test data, it was found that that in function classification of the second data test data obtained value 1 so that two classified test data fruit as category class 1 where class 1 is class Agree.

3.8 Confusion Matrix

Based on the confusion matrix, the SVM model successfully classified 232 positive and 248 negative data instances correctly. However, there were misclassifications: 62 positive data were incorrectly predicted as negative, and 58 negative data were misclassified as positive. Figure 6 illustrates this confusion matrix, providing a clear breakdown of true positives, true negatives, false positives, and false negatives.

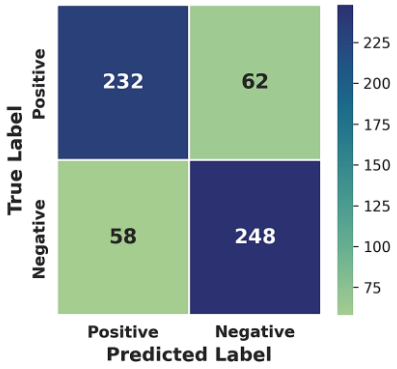

Figure 6. Confusion matrix for sentiment classification using the SVM model

The performance of the SVM classification model, including precision, recall, and F1-score for each sentiment class, is summarized in Table 24.

Table 24. Classification performance report

No	Parameter		Value
1	Accuracy		80%
2	Precision	Positive Class	80%
		Negative Class	80%
3	Recall	Positive Class	79%

TIN: Terapan Informatika Nusantara
Vol 6, No 3, August 2025, page 345-356
ISSN 2722-7987 (Media Online)
Website https://ejurnal.seminar-id.com/index.php/tin
DOI 10.47065/tin.v6i3.8210

No	Parameter		Value
4	F1-Score	Negative Class	81%
		Positive Class	79%
		Negative Class	81%

A previous study on Indonesian election sentiment using SVM collected 2,084 tweets (29.5 % positive, 70.5 % negative), achieving 73 % classification accuracy. In this study, a larger and more balanced dataset of 3,000 comments (50.67 % positive, 49.33 % negative) was used, resulting in 80 % accuracy. The improvement is attributed to the larger dataset, more comprehensive text preprocessing, and balanced class distribution. Balanced precision, recall, and F1-score values indicate that the model effectively classifies both positive and negative sentiments without significant bias, providing a more reliable analysis of public opinion.

3.9 Wordcloud

Here is the word cloud of all words in the data from the sentiment analysis of public opinion on Facebook monetization.

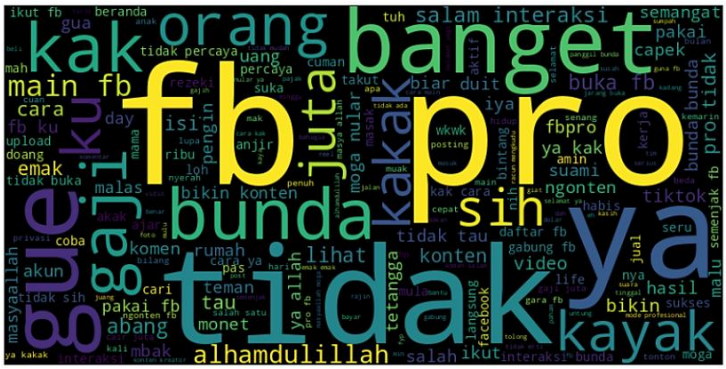

Figure 7. Word cloud of all words in the data

The overall visualization of words in the dataset, as shown in Figure 4.13, highlights several frequently appearing terms related to monetization, including “fb,” “pro,” “tidak,” “bunda,” “gaji,” “juta,” and “orang.” The frequent pairing of “tidak” with “fb” and “pro” indicates user concerns or dissatisfaction with the Facebook Pro feature and its monetization potential, rather than merely common conversational patterns. Additionally, the presence of economic terms like “gaji” and “juta” shows that users are interested in the potential earnings from the program, while other words reflect attention to account status and social interactions on the platform.

4. CONCLUSION

Based on the research results, it can be concluded that the sentiment analysis of public opinion toward Facebook’s monetization policy is relatively balanced. After preprocessing, the initial 5,000 raw comments were reduced to 3,000 clean entries. Lexicon-based labeling identified 1,520 positive comments (50.67%) and 1,480 negative comments (49.33%). The negative word cloud reflects user dissatisfaction, while the positive word cloud reveals optimism and economic benefits. Classification using a linear-kernel SVM with an 80:20 train-test split achieved an accuracy of 80%, precision of 80%, recall of 81% for negative and 79% for positive sentiments, and F1-scores of 81% and 79%, respectively. These findings contribute to the broader understanding of how policy changes in social media platforms influence user engagement and economic behavior online. Practically, the balanced sentiment distribution suggests that platform providers and content strategists should tailor monetization features to address both the technical challenges and the community’s desire for tangible rewards, thereby enhancing overall user satisfaction and long-term platform loyalty.

REFERENCES

Aguiar, L., Peukert, C., Schäfer, M., & Ullrich, H. (2022). *Facebook Shadow Profiles*. http://arxiv.org/abs/2202.04131

Aufar, A. F., Mochamad Alfan Rosid, Eviyanti, A., & Astutik, I. R. I. (2023). Optimizing Text Preprocessing for Accurate Sentiment Analysis on E-Wallet Reviews. *JICTE (Journal of Information and Computer Technology Education)*, *7*(2), 42–50. https://doi.org/10.21070/jicte.v7i2.1650

Didi, Y., Walha, A., & Wali, A. (2022). COVID-19 Tweets Classification Based on a Hybrid Word Embedding Method. *Big Data and Cognitive Computing*, *6*(2). https://doi.org/10.3390/bdcc6020058

Fadhila Sari, S., & Salsabila, I. (2024). Comparison Of K-Nearest Neighbor And Support Vector Machine For Sentiment Analysis Of The Second Covid-19 Booster Vaccination. In *Journal of Applied Statistics and Data Science*, 1(1)


TIN: Terapan Informatika Nusantara
Vol 6, No 3, August 2025, page 345-356
ISSN 2722-7987 (Media Online)
Website https://ejurnal.seminar-id.com/index.php/tin
DOI 10.47065/tin.v6i3.8210

Indriyani, F. A., Fauzi, A., & Faisal, S. (2023). Analisis sentimen aplikasi tiktok menggunakan algoritma naïve bayes dan support vector machine. *TEKNOSAINS : Jurnal Sains, Teknologi Dan Informatika*, *10*(2), 176–184. https://doi.org/10.37373/tekno.v10i2.419

Grantham, S., Cervi, L., & Iachizzi, M. (2025). Double tap democracy: political authenticity in the TikTok era. *Media International Australia*. https://doi.org/10.1177/1329878X251327232

Mohamed, N., Daud, H., P. Rameli, M. F., Man, N. C., & Mohd Aris, N. (2023). An Implemetation of Islamic Marketing Ethics among Muslimpreneurs on Digital Marketing Via Facebook. *International Journal of Academic Research in Business and Social Sciences*, *13*(9). https://doi.org/10.6007/ijarbss/v13-i9/17839

Hayadi, B. H., & Maulita, I. (2025). Sentiment Analysis of Public Discourse on Education in Indonesia Using Support Vector Machine (SVM) and Natural Language Processing. In *J. Digit. Soc*, *1(1)*

Iosifidis, P. (2025). Digital platforms and news publishers: uneasy relationship. *Frontiers in Communication*, *10*. https://doi.org/10.3389/fcomm.2025.1556826

Kah, A. El, & Zeroual, I. (2022). Sentiment analysis of students' Facebook comments toward university announcements. *2022 5th International Conference on Networking, Information Systems and Security: Envisage Intelligent Systems in 5g//6G-Based Interconnected Digital Worlds (NISS)*, 1–5. https://doi.org/10.1109/NISS55057.2022.10084994

Lisa, R. L. (2023). *Analisis Sentimen Opini Masyarakat Terhadap Institusi Polri Pada Media Sosial Twitter Menggunakan Metode Support Vector Machine Dan Naïve Bayes*. https://repository.unej.ac.id/xmlui/handle/123456789/114706

Maia, R. C., Maçada, A. C. G., & Lerch Lunardi, G. (2025). Monetization of Social Media Data: A Systematic Review of Studies, Techniques of Analysis, and Strategies for Value Creation. *Navus - Revista de Gestão e Tecnologia*, *16*, 1–22. https://doi.org/10.22279/navus.v16.1851

Markoulidakis, I., & Markoulidakis, G. (2024). Probabilistic Confusion Matrix: A Novel Method for Machine Learning Algorithm Generalized Performance Analysis. *Technologies*, *12*(7). https://doi.org/10.3390/technologies12070113

Maulana, M. R., & Putri, R. A. (2024). *Sistemasi: Jurnal Sistem Informasi Analisis Sentimen terhadap Komisi Pemilihan Umum 2024 di Indonesia melalui Twiter menggunakan Algoritma Support Vector Machine (SVM) Sentiment Analysis of the 2024 General Election Commission in Indonesia through Twiter using the Support Vector Machine (SVM) Algorithm*. http://sistemasi.ftik.unisi.ac.id

Moreno, M. A., Jolliff, A., & Kerr, B. (2021). Youth Advisory Boards: Perspectives and Processes. *Journal of Adolescent Health*, *69*(2), 192–194. https://doi.org/10.1016/j.jadohealth.2021.05.001

Mulla, S., Mandavkar, R., Jamadar, S., Magdum, S., & Gurav, U. (2025). SE_Resnet14: Design and development of deep learning architecture for kidney microscopy images grading. *Procedia Computer Science*, *259*, 1501–1510. https://doi.org/10.1016/j.procs.2025.04.105

Obi, J. C. (2023). A comparative study of several classification metrics and their performances on data. *World Journal of Advanced Engineering Technology and Sciences*, *8*(1), 308–314. https://doi.org/10.30574/wjaets.2023.8.1.0054

Rodríguez-Ibánez, M., Casánez-Ventura, A., Castejón-Mateos, F., & Cuenca-Jiménez, P. M. (2023). A review on sentiment analysis from social media platforms. In *Expert Systems with Applications*, 223, https://doi.org/10.1016/j.eswa.2023.119862

Saleh, H., Mostafa, S., Alharbi, A., El-Sappagh, S., & Alkhalifah, T. (2022). Heterogeneous Ensemble Deep Learning Model for Enhanced Arabic Sentiment Analysis. *Sensors*, *22*(10). https://doi.org/10.3390/s22103707

Shahbazi, Z., Jalali, R., & Shahbazi, Z. (2025). AI-Driven Framework for Evaluating Climate Misinformation and Data Quality on Social Media. *Future Internet*, *17*(6), 231. https://doi.org/10.3390/fi17060231

Srinidhi, K., Harika, A., Voodarla, S. S., & Abhiram, J. (2024). Sentiment Analysis of Social Media Comments using Natural Language Procesing. *2024 International Conference on Inventive Computation Technologies (ICICT)*, 762–768. https://doi.org/10.1109/ICICT60155.2024.10544849

Tangke, R., Salaki, D. T., Kalengkongan, W. W., & Ketaren, E. (2024). A*nalisis Sentimen Aplikasi Tiktok Menggunakan Algoritma Support Vector Machine (Svm) Dan Random Forest*. https://doi.org/10.51351/jtm.13.2.2024762

Tei, S., Fujino, J., & Murai, T. (2025). Navigating the self online. *Frontiers in Psychology*, 16, https://doi.org/10.3389/fpsyg.2025.1499039

Wilyani, F., Arif, Q. N., & Aslimar, F. (2024). Pengenalan Dasar Pemrograman Python Dengan Google Colaboratory. *Jurnal Pelayanan Dan Pengabdian Masyarakat Indonesia*, *3*(1), 08–14. https://doi.org/10.55606/jppmi.v3i1.1087