Available online at https://ejournal.corespub.com/index.php/ijmsc/index

International Journal of Mathematics, Statistics, and Computing

e-ISSN 3025-0803

Vol. 3, No. 3, pp. 102-112, 2025

Optimization of Machine Learning Models for Sentiment Analysis of TikTok Comment Data on the Progress of the Ibu Kota Nusantara as New Capital City of Indonesia

Renda Sandi Saputra[1*], Muhammad Bintang Eighista Dwiputra[2], Moch Panji Agung Saputra[3], Muhammad Iqbal Al-Banna Ismail[3]

[1]*Informatics Study Program, Faculty of Technology and Informatics, Universitas Informatika dan Bisnis Indonesia, Bandung, Indonesia*
[2]*Computer Science Study Program, Faculty of Mathematics and Natural Sciences Education, Universitas Pendidikan Indonesia, Bandung, Indonesia*
[3]*Department of Mathematics, Faculty of Mathematics and Natural Sciences, Universitas Padjadjaran, Sumedang, Indonesia*
[4]*School of Mathematical Sciences, Sunway University, No.5, Jalan Universiti, Bandar Sunway, 47500 Subang Jaya, Selangor, Malaysia.*

**Corresponding author email: renda.ss21@student.unibi.ac.id*

Abstract

Sentiment analysis plays a crucial role in understanding public opinion on social media platforms, especially in discussions related to government policies such as the relocation of Indonesia's new capital city, known as Ibu Kota Nusantara (IKN). While machine learning algorithms like Naïve Bayes, Support Vector Machine (SVM), and Logistic Regression (LR) are widely used for sentiment classification tasks, previous studies often focus on performance comparisons without addressing the impact of data imbalance or regularly optimizing model parameters. These issues can lead to suboptimal classification performance, especially in real-world social media data. This study aims to improve the accuracy and robustness of sentiment classification by applying two enhancement strategies: text data augmentation and hyperparameter tuning. Three models Naïve Bayes, SVM, and Logistic Regression were trained and evaluated in three experimental stages: using original data, after applying augmentation, and after augmentation combined with hyperparameter tuning via GridSearchCV. The evaluation results show progressive improvements across the three stages. In the first stage (original data), Logistic Regression achieved the highest accuracy of 80.41%, while Naïve Bayes and SVM reached 79.73% and 76.98%, respectively. However, all models struggled to classify the minority class (positive sentiment), as reflected in their lower recall and F1-scores. After applying augmentation, performance improved significantly across all models. SVM, in particular, reached an accuracy of 92.77%, followed by Logistic Regression (86.57%) and Naïve Bayes (86.22%), with better balance between precision and recall for both sentiment classes. hyperparameter tuning further optimized model performance. Logistic Regression became the best-performing model, achieving an accuracy of 93.80%, along with high precision, recall, and F1-scores for both classes. SVM and Naïve Bayes also showed stable improvements, with accuracies of 92.88% and 87.72%, respectively.

Keywords: Sentiment analysis, machine learning model, tiktok comments, data augmentation, hyperparameter tuning.

1. Introduction

The strategic plan to move the capital city from Jakarta to the Islands in East Kalimantan has been carefully planned. The planning, development, and operation of the Islands as the new capital city are regulated by a number of laws that support this initiative (Nugraha & Rido, 2025). The pros and cons of moving the National Capital are still unresolved issues in various regimes. When President Joko Widodo took concrete steps to legally move the Indonesian Capital City to IKN, the DPR approved it through Law Number 3 of 2022 concerning the Indonesian Capital City (IKN) which was passed on January 18, 2022, the public immediately responded by accepting and rejecting (Priyowidodo & Wijayanti, 2024). Despite the pros and cons, this mega project is targeted to be completed in the next 15 years (Herdiawato, 2024).

The issue of moving the National Capital City is a sensitive matter so that it is widely discussed, basically social media is used to convey opinions or an expression (Prasetyo et al., 2023). TikTok is a popular social media platform that has recently attracted a lot of attention (Fajrina et al., 2023). This platform provides a review feature that allows users to rate application services, films, e-books, and various other products. Through this feature, a large amount of data is collected in text form. The collection of text data is a rich source of information about opinions, views, and expressions of sentiment from users. However, this very large volume of reviews actually presents its own challenges in the process of effective sentiment analysis (Wahyudi & Sibaroni, 2022).

The ability of machine learning techniques to automatically process and evaluate large data sets has led to their increasing prevalence in today's technological landscape. Several different algorithms based on machine learning are used to perform sentiment analysis, and these approaches have shown promising results. Techniques such as Naive Bayes, Logistic Regression and support vector machine (SVM) are some examples of this type of methodology (Zamir et al., 2024). Various previous studies have widely applied these algorithms in the task of sentiment analysis on social media data.

Research by Assiroj et al. (2023) used data from the Twitter platform and compared the performance of three classification models, namely Naïve Bayes, Support Vector Machine (SVM), and Logistic Regression (LR), with the result that LR had the highest accuracy of 77%. The main focus of this study is on the comparison of performance between algorithms without further exploration of the aspect of improving data quality. Another study by Agresia & Suryono (2025) emphasized the importance of algorithm selection and handling data imbalance in sentiment analysis related to fraud and bot issues in concert ticket purchases. In the study, SVM showed the highest accuracy of 91.27%, followed by Logistic Regression (90.03%) and Naïve Bayes (77.70%). They also applied the Synthetic Minority Over-sampling Technique (SMOTE) to improve classification performance, especially in the negative sentiment class.

Although algorithms such as Naïve Bayes, Support Vector Machine (SVM), and Logistic Regression (LR) have been widely used in sentiment analysis research, there is still a gap in optimizing model performance through a more in-depth approach. Both previous studies tend to focus on performance comparisons between algorithms, but have not explicitly explored the application of data augmentation techniques or systematic model parameter tuning. In fact, the performance of machine learning algorithms is highly dependent on the quality and quantity of training data, as well as on the hyperparameter configuration used. Therefore, this study aims to fill this gap by applying data augmentation techniques to balance the distribution of sentiment classes, as well as performing hyperparameter tuning using the Grid Search method to improve the accuracy and efficiency of the three Naïve Bayes, SVM, and Logistic Regression models in classifying TikTok comment sentiments regarding the issue of moving the Indonesian Capital City (IKN) in Indonesia.

2. Methodology

Sentiment analysis, also known as opinion mining, is a field in natural language processing (NLP) that seeks to identify and extract subjective information from text data (Liu, 2012). With the proliferation of user-generated content on social media platforms such as TikTok, Twitter, and Facebook, sentiment analysis has become a valuable tool for gauging public opinion on contemporary issues, including political discourse, consumer behavior, and government policy implementation (Zhou et al., 2020).

TikTok, as a rapidly growing short-video platform, has evolved into more than just an entertainment space; it is now a medium through which users express public sentiment on national and political issues (Fajrina et al., 2023). The rich, real-time text content in user comments presents opportunities for analyzing collective public perspectives on high-profile issues like the development of Ibu Kota Nusantara (IKN).

Machine learning (ML) techniques have been widely adopted in sentiment classification due to their capacity to process large-scale data and learn patterns from labeled corpora (Zamir et al., 2024). Among the most common algorithms are:

- Naïve Bayes (NB): A probabilistic classifier based on Bayes' theorem, known for its simplicity and effectiveness in text classification (Ondego, 2015).
- Support Vector Machine (SVM): A discriminative classifier that finds the optimal separating hyperplane between sentiment classes by maximizing the margin (Xia, 2020).
- Logistic Regression (LR): A linear model that estimates the probability of a class label using a logistic function (Nasteski, 2017).

Each algorithm presents its own strengths and weaknesses. Previous studies, such as Assiroj et al. (2023), compared the accuracy of these algorithms using Twitter data and found that LR yielded the best accuracy at 77%, followed by SVM and NB. However, these studies often overlook deeper performance optimizations, especially related to data imbalance.

A significant challenge in sentiment analysis, particularly on social media platforms, is the imbalance of class distribution. Real-world datasets often contain more neutral or negative sentiments than positive ones (Agresia &

Suryono, 2025). This imbalance can severely impair the learning process, causing models to be biased toward the majority class.

To address this, researchers have used data-level strategies such as oversampling and augmentation. Techniques like SMOTE (Synthetic Minority Over-sampling Technique) and random word swap have been applied to enrich the minority class, improving recall and F1-score metrics (Agresia & Suryono, 2025; Behera et al., 2021). Augmentation not only improves balance but also increases the diversity and generalization capability of sentiment models.

Another essential aspect in optimizing machine learning performance is hyperparameter tuning. Each model has specific parameters that influence learning behavior, such as the penalty type in Logistic Regression, the regularization constant (C) in SVM, or the smoothing parameter (alpha) in Naïve Bayes. GridSearchCV is a popular approach for systematically exploring the optimal combination of these parameters by performing cross-validated searches (Airlangga, 2024).

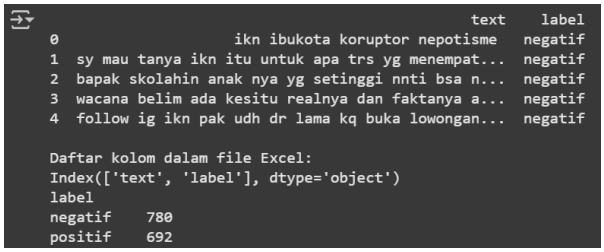
Despite the existence of these methods, few sentiment analysis studies in the Indonesian context have applied both data augmentation and hyperparameter tuning in a structured, comparative manner. Most have focused solely on algorithmic comparisons (e.g., Assiroj et al., 2023; Zamir et al., 2024), leaving a gap in comprehensive optimization strategies.

The implementation of government megaprojects, such as the Ibu Kota Nusantara (IKN) development, has sparked polarized public reactions on digital platforms. Understanding these sentiments is essential for policy feedback and social evaluation. Prasetyo et al. (2023) found that TikTok users actively participate in political discourse, especially on national matters. However, there is limited research focusing specifically on TikTok sentiment analysis in the context of IKN, highlighting the novelty and urgency of the current study.

3. Methodology

3.1. Data Collection

The data used in this study are comments from the TikTok platform discussing the development of the Indonesian Capital City (IKN). The comments were taken from two TikTok video uploads discussing the topic of IKN during the period June to July 2024.



	text	label
0	ikn ibukota koruptor nepotisme	negatif
1	sy mau tanya ikn itu untuk apa trs yg menempat...	negatif
2	bapak skolahin anak nya yg setinggi nnti bsa n...	negatif
3	wacana belim ada kesitu realnya dan faktanya a...	negatif
4	follow ig ikn pak udh dr lama kq buka lowongan...	negatif

```
Daftar kolom dalam file Excel:
Index(['text', 'label'], dtype='object')
label
negatif    780
positif    692
```

Figure 1: TikTok sentiment dataset

The dataset obtained from Kaggle contains 1,472 comments in Indonesian that have been labeled with positive and negative sentiments. Comments that only contain emojis are not included in the dataset in order to maintain data quality.

3.2. Preprocessing

The TikTok comment data that has been collected undergoes a preprocessing stage to improve text quality and reduce noise that can interfere with the analysis process. Although the data obtained from Kaggle has gone through several initial cleaning processes and is ready to be analyzed, a re-cleaning stage is still needed to ensure more optimal and accurate analysis results. The data preprocessing stages carried out are as follows:

1) Case folding
   Case folding is one of the stages in text preprocessing that functions to standardize the writing format by changing all letters in the document to lowercase. The purpose of this process is to avoid differences in interpretation of the same word due to differences in the use of capital letters (Rosid et al., 2020).
2) Text cleaning process
   This process is carried out to clean the text from non-lexical elements that do not make a meaningful contribution to sentiment analysis, such as punctuation, symbols and emojis. The presence of these elements in the text can interfere with the tokenization process and subsequent analysis, so their removal is necessary to produce a cleaner and more consistent text representation (Jazuli et al., 2024).
3) Tokenization

Tokenization is a fundamental preprocessing step to break down text into smaller yet more manageable units called tokens that facilitate machines to process and generate human language accurately. These tokens can be as small as characters or as long as words. These tokens serve as building blocks for sequences that are passed to the language model during the pre-training or fine-tuning stage (Qarah & Alsanoosy, 2024).

4) Stopword Removal
Stopwords are common words that often appear in sentences such as “a”, “this”, and the like in English. These words generally do not have a significant impact on the analysis of emotion or meaning of a sentence. Therefore, stopwords are removed so that the analysis focus is on more meaningful words. Stopword removal also helps reduce the computational load and speeds up data processing, especially when dealing with large amounts of data (Jagadishwari et al., 2021).
5) Data splitting
In this study, the augmentation process was carried out to increase the data by using the random word swap technique twice per text. From a total of 1,472 initial data, the augmentation process produced around 4,416 text data.

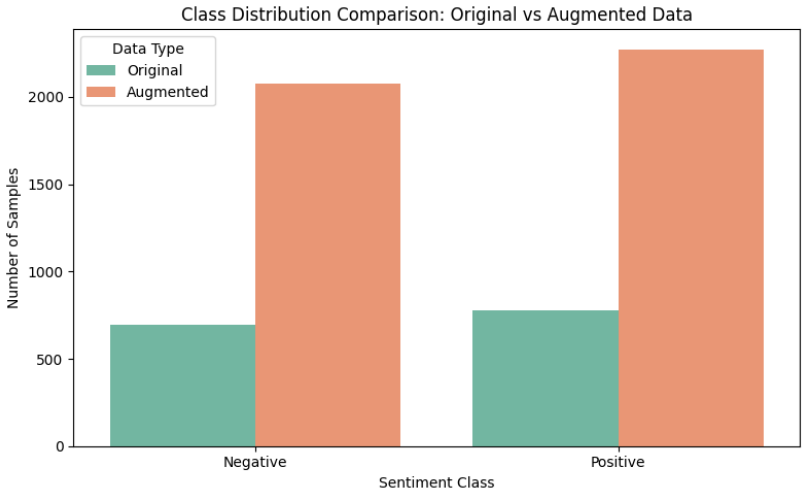


Figure 2: Total data before and after augmentation

After going through the preprocessing stage such as removing stopwords, stemming, and filtering empty text, the final amount of data used for model training and testing was 4,355 data. The data was then divided into 80% for training and 20% for testing, resulting in 871 data in the test set.

3.3. Model Building and Training

At this stage, a classification model was built for sentiment analysis using three machine learning algorithms, namely Naive Bayes, Logistic Regression, and Support Vector Machine (SVM). The model training process is divided into three phases, namely initial training (baseline), training after data augmentation, and training after augmentation and hyperparameter tuning.

1) Naive Bayes (NB)
Naive Bayes is a classification method that uses a probabilistic approach. In its application, this algorithm selects the class with the highest probability based on the features in the document or text. The class selection process is carried out using the Maximum a Posteriori (MAP) rule, where the posterior probability. Calculate how likely a comment is to fall into the "positive" or "negative" class based on the words it contains (Ondego, 2015). Bayes' theorem is used to calculate the probability of a class C based on data X with the formula:

$$P(C|X) = \frac{P(X|C) \cdot P(C)}{P(X)} \quad (1)$$

where $P(C|X)$ is the posterior probability, $P(X|C)$ is the likelihood, $P(C)$ is the prior of the class, and $P(X)$ is the total probability of the data (Suryani et al., 2023).

2) Logistic Regression (LR)
Logistic Regression is a classification model based on logistic regression, not ordinary regression. This model calculates the probability of a data falling into a certain class, then converts it into a probability value between 0 and 1

using the sigmoid function (Chang et al., 2024). Each comment is converted into a number (vector) and the Model calculates the importance weight of each word. The logistic regression hypothesis formula is defined as:

$$h_\theta(x) = g(\theta^T x) \tag{2}$$

with

$$g(z) = \frac{1}{1+e^z} \tag{3}$$

where θ is the weight parameter, x is the feature vector and $g(z) = \frac{1}{1+e^z}$ will convert the result of the linear combination $\theta^T x$ into probability (Nasteski, 2017).

3) Support Vector Machine (SVM)

SVM is a classification algorithm that finds the best boundary line (hyperplane) to separate two classes (positive and negative). This model works by maximizing the margin between data from two classes, so the prediction is more stable (Xia, 2020).

$$f(x) = \theta_0 + \sum_{j \in S} \alpha_j \cdot K(x_j, x) \tag{4}$$

The kernel function K is used to measure the similarity between the test data vector x and the support vector x_j. Parameter α_j is the parameter of the learning process results obtained from the optimization algorithm, while S is the set of support vector data, namely data that is near the margin and plays a role in forming the hyperplane (Umarani et al., 2021).

In this study, three experimental scenarios will be conducted, an experiment using original data without augmentation or tuning, to obtain the baseline performance of each model. The second experiment with augmented data, namely the addition of data variations using the Random Word Augmentation technique to increase text diversity and overcome data imbalance. Experiments with augmented data accompanied by hyperparameter tuning, namely the process of finding the best combination of parameters in each model (Naive Bayes, Logistic Regression, and SVM) using the GridSearchCV technique with the parameters tested will be shown in Table 1.

Table 1: Parameters in the GridSerchCV test

Model	Parameter	Attempted Values
Naive Bayes	alpha	[0.001, 0.01, 0.1, 0.5, 1.0]
Logistic Regression	penalty	['l1'], ['l2']
	solver	['liblinear'], ['lbfgs']
	C	[0.01, 0.1, 1.0, 10.0]
	max_iter	[1000]
SVM	C	[0.01, 0.1, 1.0, 10.0]
	loss	['hinge', 'squared_hinge']
	max_iter	[1000]

3.4. Model Evaluation

Confusion Matrix is a visual representation of statistical values obtained through an experimental process. This matrix displays statistical information about the actual and predicted labels of each review in the classified text (Behera et al., 2021). In binary classification, the confusion matrix can be presented in the form as shown in Table 2.

Table 2: Confusion matrix components

Component	Description
True Positive (TP)	Number of instances that are actually positive and correctly predicted as positive by the model.
True Negative (TN)	Number of instances that are actually negative and correctly predicted as negative by the model.
False Positive (FP)	Number of instances that are actually negative but incorrectly predicted as positive by the model.
False Negative (FN)	Number of instances that are actually positive but

	incorrectly predicted as negative by the model.

Source: Behera et al., 2021

Based on the four components in the confusion matrix, there are several evaluation metrics used to measure the classification performance of the model.

1) Accuracy
   Accuracy is the proportion of correct predictions to all tested data.

$$Accuracy = \frac{TP + TN}{TP + TN + FN + FP} \times 100\% \quad (5)$$

2) Precision
   Precision measures how many positive predictions are actually positive.

$$Precision = \frac{TP}{TP + FP} \times 100\% \quad (6)$$

3) Recall
   Recall measures how many positive data are correctly recognized by the model.

$$Recall = \frac{TP}{TP + FN} \times 100\% \quad (7)$$

4) F1-Score
   F1-score is the harmonic mean between precision and recall. F1-score is useful when a balance between precision and recall is needed, especially in imbalanced datasets (Airlangga, 2024).

$$F1 - Score = \frac{TP}{TP + FN} \times 100\% \quad (8)$$

4. Results and Discussion

4.1. Results on original data

The results of the evaluation of the Naive Bayes, Logistic Regression, and SVM models using original data will be tested to see the model performance before further testing.

Tabel 3: Classification report of models on original dataset

Model	Label	Precision	Recall	F1-Score	Accuracy
Naive Bayes	Negative	0.79	0.84	0.81	0.7973
	Positive	0.81	0.75	0.78	
	Macro Avg	0.80	0.79	0.80	
	Weighted	0.80	0.80	0.80	
Logistic Regression	Negative	0.76	0.92	0.83	0.8041
	Positive	0.88	0.68	0.77	
	Macro Avg	0.82	0.80	0.80	
	Weighted	0.82	0.80	0.80	
SVM	Negative	0.75	0.84	0.79	0.7698
	Positive	0.80	0.69	0.74	
	Macro Avg	0.77	0.77	0.77	
	Weighted	0.77	0.77	0.77	

Table 3 shows that the Logistic Regression model obtained the highest accuracy of 80.41%, followed by Naive Bayes with an accuracy of 79.73%, and SVM with the lowest accuracy of 76.98%. Naive Bayes showed relatively balanced performance between the two classes, with precision and recall each above 0.75. Meanwhile, Logistic Regression showed very good ability in detecting negative classes (recall = 0.92), but had weaknesses in identifying

positive classes (recall = 0.68), causing an imbalance in performance between classes. The SVM model produced the lowest f1-score among the three, both for negative and positive classes.

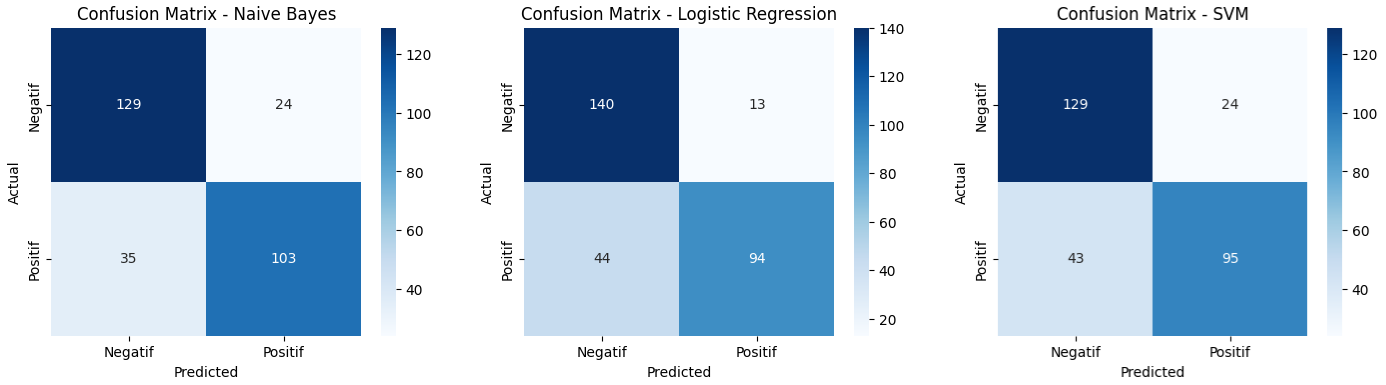


Figure 3: Confusion matrix of each model using original data

The confusion matrix results on the original data show that all three models have major challenges in recognizing the positive class. The Naive Bayes model successfully classified 103 positive data correctly, but still misclassified 35 positive data as negative. Logistic Regression excels in recognizing the negative class with 140 correct predictions, but has weaknesses in the positive class, indicated by the high false negative of 44. Meanwhile, SVM produces similar performance to Naive Bayes, with 95 correct predictions in the positive class and 43 false negatives. In general, although the accuracy is quite good, all models tend to be more accurate in predicting the negative class than the positive class.

Tabel 4: Classification report of models after data augmentation

Model	Label	Precision	Recall	F1-Score	Accuracy
Naive Bayes	Negative	0.84	0.92	0.88	0.8622
	Positive	0.90	0.80	0.85	
	Macro Avg	0.87	0.86	0.86	
	Weighted	0.87	0.86	0.86	
Logistic Regression	Negative	0.83	0.93	0.88	0.8657
	Positive	0.92	0.79	0.85	
	Macro Avg	0.87	0.86	0.86	
	Weighted	0.87	0.87	0.86	
SVM	Negative	0.91	0.96	0.93	0.9277
	Positive	0.96	0.89	0.92	
	Macro Avg	0.93	0.93	0.93	
	Weighted	0.93	0.93	0.93	

After the data augmentation process was carried out, the performance of the three models showed a significant increase compared to the previous results on the original data. Naive Bayes experienced an increase in accuracy from 79.73% to 86.22%, with an increase in f1-score in the negative class from 0.81 to 0.88, and in the positive class from 0.78 to 0.85. This model also showed an increase in precision and recall values that were more balanced between classes. In the Logistic Regression model, accuracy increased from 80.41% to 86.57%. The f1-score in the negative class increased from 0.83 to 0.88, while in the positive class it increased from 0.77 to 0.85. Although the recall of the positive class is still slightly lower than that of the negative class, the difference is smaller than before, indicating that augmentation helps reduce the inequality between classes. The SVM model showed the most significant improvement, with accuracy jumping from 76.98% to 92.77%. The F1-score for the negative class increased from 0.79 to 0.93, and for the positive class from 0.74 to 0.92. All evaluation metrics (precision, recall, and f1-score) improved consistently for both classes.

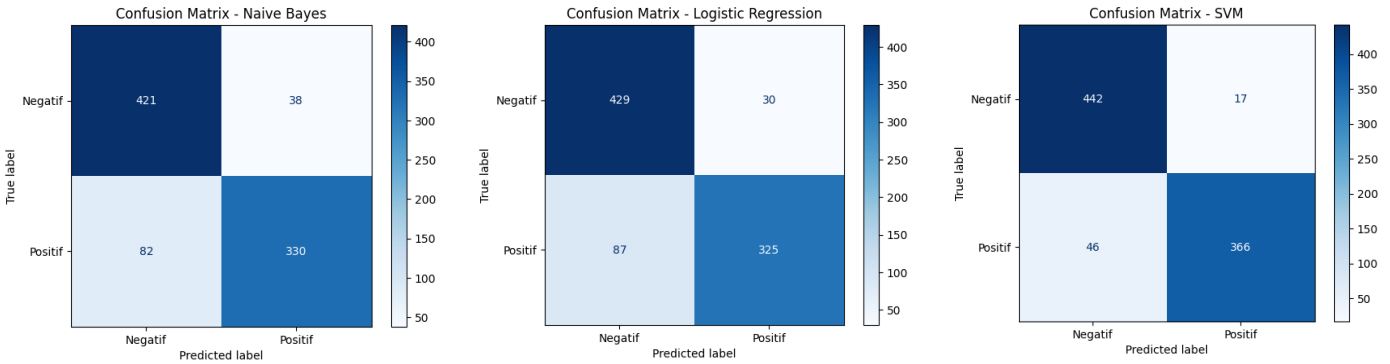


Figure 4: Confusion matrix of each model after data augmentation

The performance of the three models improved after data augmentation. The Naive Bayes model successfully classified 330 positive data correctly, an increase from the previous (103), and only produced 82 false negatives, indicating an increase in the ability to recognize the positive class. The Logistic Regression model also experienced significant improvement, with 325 true positives and 87 false negatives, although slightly higher than Naive Bayes. Meanwhile, the SVM model showed the best results, with 366 correct predictions for the positive class and only 46 false negatives, and very few errors in the negative class (only 17 false positives).

Table 5: Evaluation results with the addition of Augmentation and GridSearchCV

Model	Label	Precision	Recall	F1-Score	Accuracy
Naive Bayes	Negative	0.86	0.92	0.89	0.8772
	Positive	0.90	0.83	0.86	
	Macro Avg	0.88	0.87	0.88	
	Weighted	0.88	0.88	0.88	
Logistic Regression	Negative	0.92	0.97	0.94	0.9380
	Positive	0.96	0.90	0.93	
	Macro Avg	0.94	0.94	0.94	
	Weighted	0.94	0.94	0.94	
SVM	Negative	0.92	0.95	0.93	0.9288
	Positive	0.94	0.91	0.92	
	Macro Avg	0.93	0.93	0.93	
	Weighted	0.93	0.93	0.93	

After hyperparameter tuning using GridSearchCV on the augmented data, the performance of all models increased compared to the previous two conditions. In the Naive Bayes model, the accuracy increased from 79.73% (original data) and 86.22% (augmentation) to 87.72% after tuning. The f1-score for the negative class increased from 0.88 to 0.89, and for the positive class from 0.85 to 0.86. Although the increase was not as large as after augmentation, tuning helped improve the balance of performance between classes, especially in recall and macro average.

The Logistic Regression model showed the most significant improvement. The initial accuracy of 80.41% (original data) and 86.57% (augmentation) increased sharply to 93.80% after tuning. The f1-score for the negative class increased from 0.88 to 0.94, and for the positive class from 0.85 to 0.93. This shows that tuning not only improves overall performance, but also substantially improves precision and recall on both classes, especially in recognizing the previously more difficult positive class. The SVM model also improved, although not as much as Logistic Regression. From 76.98% (original data) and 92.77% (augmentation) accuracy, it increased to 92.88% after tuning. The F1-score for the negative class increased from 0.93 to 0.93 (stable), and for the positive class from 0.92 to 0.92. This shows that the SVM is already highly optimized after augmentation, and tuning provides small improvements but still maintains high and stable performance.

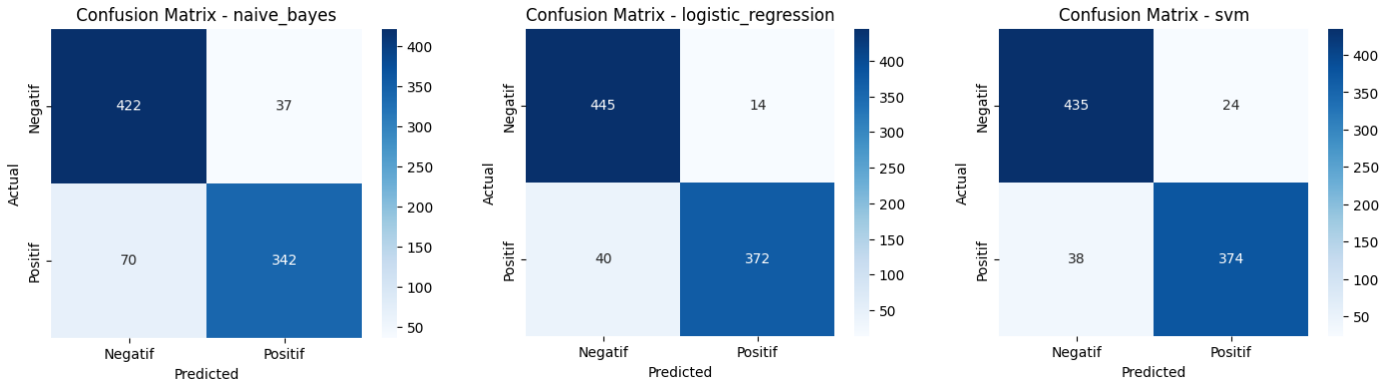


Figure 5: Confusion matrix of each model after data augmentation and hyperparameter tuning

All three models achieved their best performance after augmentation and tuning. The Naive Bayes model recorded 342 true positives and only 70 false negatives, and 422 true negatives with 37 false positives, showing an improvement from before, especially in the ability to recognize the positive class. The Logistic Regression model became the most accurate model, with 372 positive data correctly classified and only 40 false negatives, and very few errors in the negative class (445 true negatives, only 14 false positives). Meanwhile, the SVM model also showed very good results, with 374 true positives and only 38 false negatives, and 435 true negatives and 24 false positives. After hyperparameter tuning using GridSearchCV, each model obtained the best parameter configuration that could significantly improve performance.

Tabel 6: Best parameters obtained from GridSearchCV for each model

Model	Parameter	Value
Naive Bayes	alpha	0.01
Logistic Regression	C	10.0
	max_iter	1000
	penalty	'l1'
	solver	'liblinear'
SVM	C	10.0
	loss	'squared_hinge'
	max_iter	1000

For the Naive Bayes model, the best parameter obtained was alpha = 0.01, which functions as a smoothing value to avoid division by zero and to set the strength of regularization; this small value indicates that the model does not require large smoothing because the data is more balanced after augmentation. In the Logistic Regression model, the best parameter combination is C = 10.0, penalty = 'l1', solver = 'liblinear', and max_iter = 1000. A large C value gives significant weight to the training data, and the use of l1 penalty is suitable for producing simpler models with automatically selected features (sparse). The liblinear solver supports l1 penalty with high efficiency. For the SVM model, the optimal parameters are C = 10.0, loss = 'squared_hinge', and max_iter = 1000. A large C value indicates that the model penalizes misclassifications highly, while squared_hinge is used as a common loss function in linear SVM implementations, providing a more stable margin during training.

5. Conclusion

1) Model Evaluation on Original Data
   Initial testing was conducted on original data without modification, using three classification algorithms: Naive Bayes, Logistic Regression, and Support Vector Machine (SVM). The evaluation results show that all models tend to have better performance in recognizing the negative class, while the performance on the positive class is still low, especially in terms of recall. Logistic Regression recorded the highest accuracy of 80.41%, followed by Naive Bayes (79.73%) and SVM (76.98%). The confusion matrix indicates that the number of False Negatives is still high, which shows the challenge in accurately identifying the minority class.
2) Performance Improvement After Data Augmentation
   After the text augmentation process, all three models experienced significant performance improvements. The previously imbalanced data became more representative, so the models were able to recognize patterns from both classes better. Naive Bayes increased its accuracy to 86.22%, Logistic Regression to 86.57%, and SVM

increased sharply to 92.77%. In addition to accuracy, improvements are also seen in the precision, recall, and f1-score metrics, especially for the positive class. The confusion matrix shows a reduction in false negatives and false positives, indicating that augmentation has succeeded in improving the balance of predictions between classes.

3) Model Optimization Through Hyperparameter Tuning
The final step of hyperparameter tuning using GridSearchCV provides increasingly optimal results. The best parameters were found for each model, such as alpha = 0.01 for Naive Bayes, and C = 10.0, penalty = 'l1', solver = 'liblinear' for Logistic Regression. The results of this tuning have a positive impact on all evaluation metrics. Logistic Regression is the best model with a final accuracy of 93.80%, followed by SVM (92.88%) and Naive Bayes (87.72%). These results show that the combination of data augmentation and parameter tuning is a very effective strategy in improving the performance of sentiment classification models, both in terms of accuracy and the ability to recognize both classes equally.

References

Agresia, V., & Suryono, R. R. (2025). Comparison of SVM, Naïve Bayes, and Logistic Regression Algorithms for Sentiment Analysis of Fraud and Bots in Purcashing Concert Ticket. *INOVTEK Polbeng-Seri Informatika*, *10*(2), 591-602.

Airlangga, G. (2024). Optimizing SMS spam detection using machine learning: A comparative analysis of ensemble and traditional classifiers. *Journal of Computer Networks, Architecture and High Performance Computing*, *6*(4).

Assiroj, P., Kurnia, A., & Alam, S. (2023). The performance of Naïve Bayes, support vector machine, and logistic regression on Indonesia immigration sentiment analysis. *Bulletin of Electrical Engineering and Informatics*, *12*(6), 3843-3852.

Behera, R. K., Jena, M., Rath, S. K., & Misra, S. (2021). Co-LSTM: Convolutional LSTM model for sentiment analysis in social big data. *Information Processing & Management*, *58*(1), 102435.

Chang, V., Sivakulasingam, S., Wang, H., Wong, S. T., Ganatra, M. A., & Luo, J. (2024). Credit Risk Prediction Using Machine Learning and Deep Learning: A Study on Credit Card Customers. *Risks*, *12*(11), 174.

Fajrina, D. P., Syafriandi, S., Amalita, N., & Salma, A. (2023). Sentiment Analysis of TikTok Application on Twitter using The Naïve Bayes Classifier Algorithm. *UNP Journal of Statistics and Data Science*, *1*(5), 392-398.

Herdiawanto, H. (2023). The position of the Nusantara Capital City from a national security perspective Kedudukan Ibu Kota Nusantara dalam perspektif keamanan nasional. *Masyarakat, Kebudayaan dan Politik*, *36*(4), 545-558.

Jagadishwari, V., Indulekha, A., Raghu, K., & Harshini, P. (2021, November). Sentiment analysis of social media text-emoticon post with machine learning models contribution title. In *Journal of Physics: Conference Series* (Vol. 2070, No. 1, p. 012079). IOP Publishing.

Jazuli, A., Widowati, & Kusumaningrum, R. (2024). Optimizing Aspect-Based Sentiment Analysis Using BERT for Comprehensive Analysis of Indonesian Student Feedback. *Applied Sciences*, *15*(1), 172.

Liu, B. (2012). *Sentiment Analysis and Opinion Mining*. Morgan & Claypool Publishers.

Nasteski, V. (2017). An overview of the supervised machine learning methods. *Horizons. b*, *4*(51-62), 56.

Nugraha, R., & Rido, A. (2025). Nusantara: A New Capital City or a Costly Gamble for Indonesia?. *Ranah Research: Journal of Multidisciplinary Research and Development*, *7*(4), 2588-2594.

Ondego, G. (2015). *A comparative study of decision tree and naïve bayesian classifiers on verbal autopsy datasets* (Doctoral dissertation, University of Nairobi).

Prasetyo, S. D., Hilabi, S. S., & Nurapriani, F. (2023). Sentiment Analysis of the Relocation of Nusantara Capital City Using Naïve Bayes and KNN Algorithms. *Jurnal KomtekInfo*, 1-7.

Priyowidodo, G., & Wijayanti, C. A. (2024). *Public opinion on Indonesias capital relocation policy: a netnographic analysis* (Doctoral dissertation, Petra Christian University).

Qarah, F., & Alsanoosy, T. (2024). A comprehensive analysis of various tokenizers for arabic large language models. *Applied Sciences*, *14*(13), 5696.

Rosid, M. A., Fitrani, A. S., Astutik, I. R. I., Mulloh, N. I., & Gozali, H. A. (2020, June). Improving text preprocessing for student complaint document classification using sastrawi. In *IOP Conference Series: Materials Science and Engineering* (Vol. 874, No. 1, p. 012017). IOP Publishing.

Suryani, S., Fayyad, M. F., Savra, D. T., Kurniawan, V., & Estanto, B. H. (2023). Sentiment Analysis of Towards Electric Cars using Naive Bayes Classifier and Support Vector Machine Algorithm. *Public Research Journal of Engineering, Data Technology and Computer Science*, *1*(1), 1-9.

Wahyudi, D., & Sibaroni, Y. (2022). Deep learning for multi-aspect sentiment analysis of tiktok app using the rnn-lstm method. *Building of Informatics, Technology and Science (BITS)*, *4*(1), 169-177.

Xia, Y. (2020). Correlation and association analyses in microbiome study integrating multiomics in health and disease. *Progress in molecular biology and translational science*, *171*, 309-491.

Zamir, M. T., Ullah, F., Tariq, R., Bangyal, W. H., Arif, M., & Gelbukh, A. (2024). Machine and deep learning algorithms for sentiment analysis during COVID-19: A vision to create fake news resistant society. *PloS one*, *19*(12), e0315407.

Zhou, Q., Chen, Y., & Huang, S. (2020). *Sentiment analysis in the age of social media*. Computational Social Science Review.