Available online at website: https://jurnal.iaii.or.id/index.php/RESTI

JURNAL RESTI
(Rekayasa Sistem dan Teknologi Informasi)
Vol. 9 No. 4 (2025) 905 - 914 e-ISSN: 2580-0760

Sentiment Analysis Optimization Using Ensemble of Multiple SVM Kernel Functions

M. Khairul Anam[1]*, Tri Putri Lestari[2], Lusiana Efrizoni[3], Nadya Satya Handayani[4], Imam Andhika[5]
[1]Department of Informatics, Faculty of Science and Technology, Universitas Samudra, Langsa, Indonesia
[2]Department of Business Digital, Faculty of Education and Social Sciences, Universitas Indraprasta PGRI, Jakarta, Indonesia
[3]Department of Informatics Engineering, Universitas Sains dan Teknologi Indonesia, Pekanbaru, Indonesia
[4]Department of Informatics Engineering, Politeknik Negeri Batam, Batam, Indonesia
[5]Department of Informatics, Faculty of Mathematics and Natural Sciences, Universitas Syiah Kuala, Banda Aceh, Indonesia

[1]khairulanam@unsam.ac.id, [2]tplestari89@gmail.com, [3]lusiana@usti.ac.id, [4]nadya.satya007@gmail.com, [5]imamandhika14@usk.ac.id

Abstract

This study targets improved sentiment classification by combining the strengths of multiple SVM kernels within an ensemble framework. We introduce SVM Porlis, which fuses Linear, RBF, Polynomial, and Sigmoid kernels using both hard and soft voting to boost performance on skewed data. The task is binary sentiment recognition (positive vs. negative). A corpus of 2,248 tweets concerning the debate over the naturalization of Indonesia's national football players was gathered via the official X/Twitter API, with a marked dominance of negative tweets. The preprocessing pipeline encompassed cleaning, labeling, tokenization, stopword removal, stemming, and TF-IDF feature extraction. To counter the imbalance, SMOTE was applied to synthesize additional minority-class samples. Each kernel was first trained and assessed independently, then aggregated into the SVM Porlis ensemble. Evaluation used accuracy, precision, recall, F1-score, and confusion-matrix analysis. The soft-voting SVM Porlis model achieved the best results—98% for accuracy, precision, recall, and F1—outperforming single-kernel baselines and other ensembles such as SVM + Chi-Square and SVM + PSO. These outcomes indicate that integrating diverse kernels effectively captures both linear and nonlinear patterns, yielding a robust and adaptive approach for sentiment analysis on real-world, imbalanced datasets.

Keywords: ensemble learning; kernel function; sentiment analysis; smote; support vector machine

How to Cite: M. Khairul Anam, T. P. Lestari, L. Efrizoni, N. S. Handayani, and I. Andhika, "Sentiment Analysis Optimization Using Ensemble of Multiple SVM Kernel Functions", *J. RESTI (Rekayasa Sist. Teknol. Inf.)* , vol. 9, no. 4, pp. 905 - 914, Aug. 2025.
Permalink/DOI: https://doi.org/10.29207/resti.v9i4.6708

Received: June 2, 2025
Accepted: July 23, 2025
Available Online: August 22, 2025

This is an open-access article under the CC BY 4.0 License
Published by Ikatan Ahli Informatika Indonesia

1. Introduction

Support Vector Machine (SVM) is a widely adopted Machine Learning technique noted for its robust performance in classification as well as regression [1]. Renowned for its accuracy, SVM excels by maximizing the margin, enabling it to perform well even with limited data [2]. Additionally, SVM is versatile, utilizing different types of kernels such as linear, polynomial, and radial basis function (RBF), which allows it to handle both linear and non-linear data by mapping it into higher-dimensional spaces [3].

However, SVM has notable weaknesses worth addressing. Firstly, it may face computational complexity challenges with large datasets, as its time complexity heavily depends on the size of the data [4]. The training process can be especially time-consuming when there is a large number of samples and features [5]. Secondly, SVM struggles to perform effectively on noisy datasets or when classes overlap, as it is sensitive to outliers that may distort the hyperplane margin [6]. Another limitation is the dependency on selecting the correct kernel; choosing an inappropriate kernel can significantly decrease model accuracy. Furthermore, tuning kernel parameters often requires expertise and extensive experimentation [7].

Several prior studies have proposed improvements to the Support Vector Machine (SVM) algorithm for sentiment analysis. One study enhanced SVM performance by implementing chi-square, hyperparameter GridSearch, and Bagging, achieving an accuracy of 93.40% [8]. Another introduced GBSVM,

a hybrid of SVM and Gradient Boosting, which reached a predictive accuracy of 93% [9]. The combination of chi-square and SVM also showed promising results, achieving an accuracy of 95.56% [10]. Other ensemble methods combining SVM with different algorithms have demonstrated strong performance, with accuracy reaching 96.53% [11]. Studies involving Indonesian-language datasets also vary in performance. For instance, a study using Random Forest with Uncertainty Sampling on Indonesian Twitter data achieved 81% accuracy [12], while another used Naïve Bayes to analyze Instagram comments about Anies Baswedan and achieved 85%, though it suffered from overfitting issues [13]. Further, a study incorporating SMOTE improved the performance of algorithms like Logistic Regression, SVM, and Naïve Bayes, reaching up to 89% accuracy [14].

While these studies highlight the effectiveness of ensemble techniques, chi-square, and hyperparameter optimization in enhancing SVM, a critical gap remains: few have explored or reported the specific impact of different SVM kernel functions on sentiment classification. Yet, kernel selection is a vital component in SVM modeling, especially for sentiment analysis tasks that involve non-linear, noisy, and imbalanced data with complex subjective expressions. This study addresses this gap by systematically evaluating four core SVM kernels—RBF, Linear, Polynomial, and Sigmoid—using Indonesian-language tweets collected via the X (Twitter) API.

SVM offers several kernel options such as Linear, Polynomial, RBF, Gaussian, Gaussian-Diagonal, Laplace_rbf, Anova_rbf, and Sigmoid [15]. In this study, only four kernels (RBF, Linear, Polynomial, and Sigmoid) were selected for comparison and integration to improve performance on unstructured data. The RBF (Radial Basis Function) kernel is particularly effective in handling non-linear decision boundaries, which are common in sentiment data, and provides a strong structure for generalization [16]. Its gamma parameter controls the sensitivity to the distance between data points, helping capture subtle emotional nuances [17]. Previous studies have shown that this kernel achieves high accuracy in sentiment analysis, reaching 87.25% [18]. It is also highly effective for structured data and has demonstrated strong performance in prior research with an accuracy of 93.55% [19]. The Polynomial kernel handles complex relationships well, especially when sentiment expressions involve interactions among multiple features. Its degree parameter allows control over model complexity [20], , and it has shown good performance in sentiment analysis with an accuracy of 84% [21]. The Sigmoid kernel, which resembles neural network activation functions, is suitable for modeling moderate non-linear patterns. Although not as widely used as other kernels, it remains effective for datasets with intermediate complexity [22]. In sentiment analysis, it has shown exceptional performance, achieving an accuracy of up to 96.26% [23].

To further improve classification accuracy and robustness, this research proposes an ensemble model named SVM Porlis, which integrates the four kernels using both hard and soft voting techniques. This approach is designed to maximize the unique capabilities of each kernel: Linear for linear discrimination [24], RBF for capturing highly non-linear structures [25], Polynomial for multi-way non-linear feature interactions [26], and Sigmoid for added flexibility and adaptability [27]. The combination of these kernels enables the model to generalize across diverse, imbalanced, and noisy sentiment datasets—common characteristics in social media texts. In this ensemble, voting mechanisms play a crucial role in mitigating the limitations of individual kernels. When one or more base classifiers exhibit relatively lower accuracy on certain instances, the final prediction is not determined solely by that weak performer. Instead, hard voting considers the majority prediction from all kernels, while soft voting aggregates the predicted probabilities and selects the class with the highest combined confidence. This ensures that even if a single kernel underperforms in specific scenarios, the ensemble decision still benefits from the strengths of the more accurate kernels, resulting in improved overall performance and stability.

In the sentiment classification process, the study utilized tweet data obtained through the official API of the X/Twitter platform, totaling 2,248 entries related to the issue of naturalization of Indonesian national football players. The data was categorized into two sentiment classes: positive and negative. However, the class distribution in the dataset was imbalanced, with the number of tweets expressing negative sentiment significantly higher than those with positive sentiment. To address this issue, the Synthetic Minority Over-sampling Technique (SMOTE) was applied to strengthen the representation of the minority class, allowing the model to generalize more effectively across both classes. The preprocessing stage included cleaning irrelevant characters or symbols, labeling the data, tokenizing to split sentences into individual words, removing stopwords, and performing stemming to return words to their root form. The processed data was then represented using the Term Frequency–Inverse Document Frequency (TF-IDF) method, which transforms textual data into high-dimensional numerical features that can be processed by the model. For model construction, four kernels from the Support Vector Machine (SVM) algorithm—Linear, RBF, Polynomial, and Sigmoid—were employed and combined into an ensemble model named SVM Porlis, utilizing both soft voting and hard voting techniques to enhance classification performance.

This study is expected to leverage the strengths of each kernel within the SVM algorithm to develop a model that excels in terms of accuracy, precision, recall, and F1-score. The proposed model, named SVM Porlis, is designed to demonstrate strong generalization

capabilities and high reliability when applied to real-world sentiment analysis scenarios, particularly in processing complex and unstructured Indonesian-language social media data. This approach is anticipated to make a meaningful contribution as a practical and adaptive solution in the development of SVM-based sentiment classification models. The research specifically focuses on binary classification, distinguishing between two main sentiment classes—positive and negative—which are commonly used in public opinion analysis on social media platforms.

2. Methods

In this study, the effectiveness of the SVM algorithm with four of its kernels, namely, Linear, RBF, Polynomial and Sigmoid has been confirmed. The kernels are therefore combined into a model called SVM Porlis which is basically designed to take up all the advantages of all the kernels. The constructed model is considered to be as per the design depicted in Figure 1.

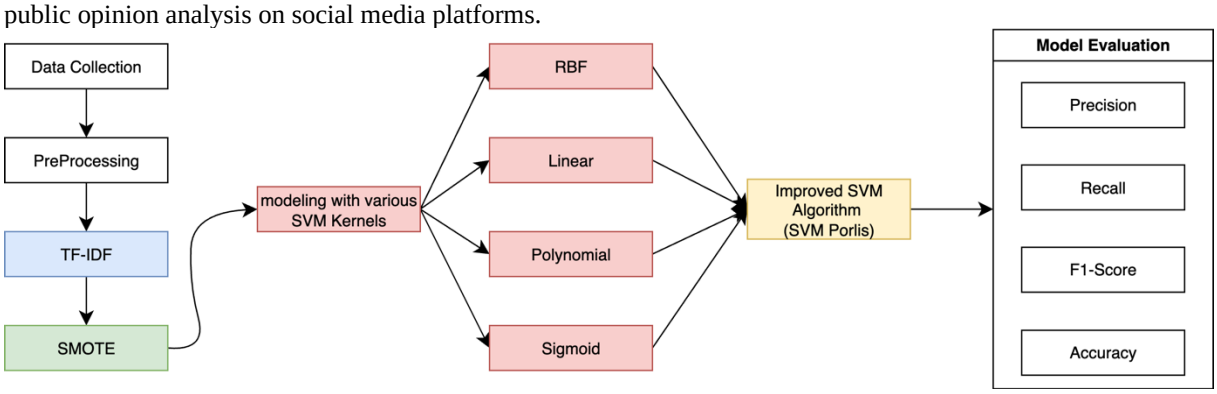


Figure 1. Research Methodology

2.1 Dataset and Preprocessing

Data preprocessing is a foundational step in sentiment analysis because it converts raw, noisy text into well-structured inputs that machine-learning models can effectively consume [28]. In this study, tweets were gathered in November 2024 via the official X/Twitter API using an open-source harvester available on GitHub (https://github.com/fikrimln16/data-crawling-x-tweetharvest) [29]. The collection window was aligned to the period immediately before and after the Indonesia vs. Japan match in the 2026 FIFA World Cup Qualifiers. A total of 2,248 tweets written in Bahasa Indonesia were gathered, capturing public discourse around the issue of naturalization of Indonesian national football team players. These tweets represent genuine user opinions and commonly contain noise such as informal expressions, abbreviations, slang, emojis, hashtags, and symbols—typical characteristics of social media text.

The corpus was labeled into two sentiment classes—positive and negative—but the distribution was highly skewed, with negative tweets greatly outnumbering positives (see Figure 2). To mitigate this imbalance, we applied the Synthetic Minority Oversampling Technique (SMOTE) to generate synthetic examples for the minority (positive) class, producing a more balanced training set and improving generalization across both classes.

The preprocessing pipeline involved several steps. First, cleaning was performed to remove irrelevant content such as punctuation marks, digits, hyperlinks, HTML tags, and emojis to produce cleaner input suitable for feature extraction [30]. The next step was labeling, in which tweets were manually annotated into binary sentiment labels—positive and negative—forming the ground truth required for supervised learning [31]. Following labeling, the text was tokenized, breaking it down into smaller units (tokens), typically words or phrases, which simplifies the modeling process [32].

To further refine the input, stopword removal was carried out by eliminating common yet semantically weak words in Bahasa Indonesia, such as “dan,” “yang,” and “di,” which occur frequently but contribute little to sentiment differentiation [33]. This was followed by stemming using the Sastrawi stemmer, an NLP library for Bahasa Indonesia, which reduces each word to its root form—for example, “bermain” is reduced to “main”—to reduce vocabulary dimensionality while preserving meaning [34].

To further illustrate the impact of the preprocessing pipeline, Table 1 presents several examples of tweet text before and after preprocessing. These samples demonstrate how noisy, informal, and unstructured social media content is transformed into clean, tokenized, and semantically meaningful input suitable for machine learning models.

Table 1. Classification Report using SVM Porlis Soft Voting

Original Tweet	After Preprocessing
@TimnasGaruda bermain sangat buruk 😡! #INDvsJPN	timnas garuda main buruk
Wah, pemain naturalisasi cuma jalan doang di lapangan, ngapain diambil???	pemain naturalisasi jalan lapangan ngapain ambil
Keren banget! Pemain baru kita cetak gol!!🔥	keren pemain baru cetak gol
Fix deh, naturalisasi hanya buat gagalin pemain lokal 😒	naturalisasi gagal pemain lokal
Salut buat perjuangan timnas meskipun kalah. Tetap semangat!	salut perjuangan timnas kalah semangat

These examples demonstrate the conversion of raw tweets into standardized input by removing hashtags, emojis, punctuation, and stopwords, and by reducing each word to its root form using stemming. Such transformations improve the quality of features passed to the classifier, which in turn contributes to the robustness and accuracy of the sentiment analysis model.

After normalization, each review was represented using the Term Frequency–Inverse Document Frequency (TF-IDF) scheme, which converts text into high-dimensional sparse numeric vectors. TF-IDF assigns larger weights to terms that are frequent within a document yet infrequent across the corpus, thereby highlighting tokens most informative for sentiment classification [35]. The formulations of term frequency, inverse document frequency, and their product (TF-IDF) used in this study are provided in Equations 1–3.

Term Frequency (TF):

$$TF = \frac{\text{Number of times the word appears in the document}}{\text{Total words in the document}} \quad (1)$$

Inverse Document Frequency (IDF):

$$IDF = \frac{Total\ number\ of\ document}{number\ of\ document\ containing\ the\ word} \quad (2)$$

TF-IDF Score:

$$TF - IDF = TF \times IDF \quad (3)$$

These TF-IDF features serve as input to the classification model. Each unique word becomes a feature column in the resulting matrix, with its TF-IDF weight populating the corresponding cell. Terms with higher weights are prioritized by the model, while common terms such as conjunctions and prepositions are down-weighted, improving the classifier's focus on sentiment-bearing words.

The complete preprocessing pipeline—spanning from data collection to TF-IDF vectorization—forms a foundational element in this research, ensuring that the SVM Porlis ensemble classifier receives clean, balanced, and discriminative input to optimize its sentiment classification performance.

2.2 SMOTE

This research addresses the problem of class imbalance in the sentiment dataset, which can cause machine learning models to be biased toward the majority class. Figure 2 presents the initial distribution of sentiment labels prior to balancing. It clearly illustrates that the majority class (label 0) significantly outnumbers the minority class (label 1), a condition that may lead to poor performance in detecting underrepresented but often critical sentiment patterns such as negative opinions.

To address the class-imbalance problem, we employed the Synthetic Minority Oversampling Technique (SMOTE). This method synthesizes additional minority-class examples by creating points along the line segments that connect a sample to its nearest minority neighbors, thereby expanding the minority class and mitigating imbalance [30]. After SMOTE is applied, the label distribution becomes balanced as shown in Figure 3 with both classes containing an equal number of instances.

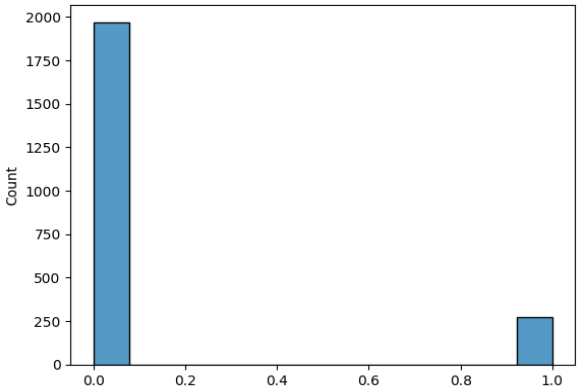


Figure 2. Data Before Class Balancing

However, applying SMOTE requires careful consideration, especially regarding its timing in the preprocessing pipeline [36]. If SMOTE is applied before splitting the dataset into training and testing sets, there is a risk of overfitting due to synthetic samples leaking into both subsets. This contamination can lead to overly optimistic performance estimates, as the model may encounter nearly identical synthetic samples during both training and testing [37]. To mitigate this risk, in this study, SMOTE was strictly applied only to the training set after data splitting, ensuring that the test set remained purely representative of real-world data.

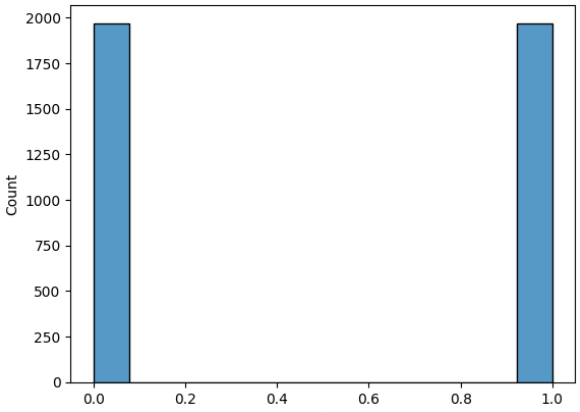


Figure 3. Data After Class Balancing Process

Besides balancing, this study also implemented strategies to prevent overfitting and ensure that the model generalizes well to unseen data. The dataset was divided into training and testing sets using a stratified split, preserving the proportion of classes across both subsets to avoid skewed learning. Regularization parameters (C, gamma) were fine-tuned to control the complexity of individual SVM models and prevent them from memorizing training data. Moreover, the use of an ensemble approach—through hard and soft voting

in the SVM Porlis model—reduces the risk of overfitting by aggregating predictions from diverse kernels. This ensemble mechanism stabilizes the model's decision boundary and helps it perform reliably across varying sentiment patterns.

In summary, class balancing through SMOTE and the use of ensemble voting, combined with proper data splitting and regularization, collectively contribute to the robustness and generalizability of the sentiment classification model [38].

2.3. Modeling with SVM

This work trains SVM models with four kernel functions. The linear kernel is appropriate when the classes are linearly (or nearly linearly) separable. It computes similarity as the inner product of two feature vectors [39] using Equation 4.

$$K(x_i, x_j) = x_i . x_j \quad (4)$$

Linear models are often adequate for straightforward sentiment tasks where feature relationships are close to linear. The most influential hyperparameter is C, which controls regularization: larger C reduces bias but can raise variance.

Radial Basis Function (RBF) kernel. The RBF kernel is effective for capturing non-linear structure that frequently appears in sentiment data. It measures similarity with a Gaussian function [40], see (5).

$$K(x_i, x_j) = \exp(-\gamma ||x_i - x_j||^2) \quad (5)$$

The γ parameter sets the radius of influence of individual training points (small γ = Narrow). As with the linear kernel, C regulates the balance between fit and generalization.

Polynomial kernel. This kernel extends the linear case by introducing polynomial interactions among input features, enabling the model to represent higher-order non-linear boundaries [41] as expressed in Equation (6). Key hyperparameters include C, the degree of the polynomial, and coef0; higher degrees allow more complex decision surfaces but increase overfitting risk.

$$K(x_i, x_j) = (x_i . x_j + c)^d \quad (6)$$

Where c is a constant term (also known as coef0) and d is the degree of the polynomial. The degree controls the flexibility of the decision boundary. In this study, we tested various degrees and selected optimal values during initial experimentation to balance complexity and performance.

Finally, the Sigmoid kernel mimics the behavior of a neural network activation function, making it useful in modeling certain non-linear relationships [22] as shown in Equation 7.

$$K(x_i, x_j) = tanh(\alpha(x_i . x_j) + c) \quad (7)$$

Here, α (alpha) is the scaling parameter and c is the bias term. These hyperparameters affect the curvature of the decision boundary. While less commonly used than RBF, the Sigmoid kernel is included in this study to evaluate its adaptability to moderately complex sentiment patterns.

2.4 Modeling with SVM Porlis

Upon conclusion of the analysis and comprehension of the data through distinct SVM kernels (Linear, RBF, Polynomial, and Sigmoid), ensemble voting techniques were applied to enhance prediction accuracy. This approach combines the strengths of different kernel perspectives. Voting was implemented in two ways: hard voting and soft voting.

In hard voting, the final class label is determined by the majority class predicted by all base models. The decision rule is as shown in Equation 8.

$$\mathrm{Y} = \mathrm{argmax}(\textstyle\sum_{i=1}^{n} I(y_i = c)) \quad (8)$$

$I(y_i = c)$ is the Indicator that is 1 if model i predicts class c, 0 otherwise. n is the Number of models used in voting.

In soft voting, each base model outputs a probability distribution over classes. The final class is selected based on the highest average probability:

$$\mathrm{Y} = \mathrm{argmax}\left(\tfrac{1}{n}\textstyle\sum_{i=1}^{n} P_i(c)\right) \quad (9)$$

$P_i(c)$ is the Prediction probability of class c from model i. n is the Number of models used.

In the implementation, soft voting was enabled by setting probability=True in each SVM model to allow probability outputs. This technique enhances model flexibility by considering confidence levels from each base classifier, thereby offering improved robustness compared to hard voting, especially in borderline cases.

From Equations 8 and 9, the next is Pseudocode1. In this process, the data that has been balanced with SMOTE is trained using four SVM models with different kernels (Linear, RBF, Polynomial, Sigmoid). After that, the predictions from each model are combined using two voting techniques, namely hard voting and soft voting. In hard voting, the selected class is the class that is most frequently predicted by all models. In contrast, in soft voting, the prediction probabilities from all models are combined, and the class with the highest probability is selected as the final result. This technique allows each kernel to contribute to its strengths, thus improving the accuracy and robustness of the model to various data patterns. The end result is an ensemble model that is more robust and flexible than using a single kernel.

To ensure transparency and reproducibility, the specific kernel parameters used in each SVM model were configured as: For the Linear kernel, the penalty parameter *C* was set to 1.0; For the RBF kernel, *C* = 1.0 and γ ='scale', which automatically adjusts gamma to the number of features; For the polynomial kernel, parameter were set to *C* = 1.0, degree = 3, and coef0 =1;

For the Sigmoid kernel, C = 1.0 and coef0 = 0 were used.

These settings were based on empirical tuning and maintained consistently across all base models in the ensemble.

```
Pseudocode 1. SVM Porlis
Import required libraries:
  - Import `SVC` from sklearn.svm and
    `VotingClassifier`from
    sklearn.ensemble.
Preprocess the data:
  - Balance the dataset using SMOTE.
  - Split the dataset into training and
    testing sets.
Train individual SVM models with different
kernels:
  - Define and train SVM with Linear
    kernel.
  - Define and train SVM with RBF kernel.
  - Define and train SVM with Polynomial
    kernel.
  - Define and train SVM with Sigmoid
    kernel.
Combine the models using Voting:
  - For hard voting:
    a. Initialize `VotingClassifier` with
       `voting='hard'`.
    b. Include all trained SVM models as
       estimators.
  - For soft voting:
    a. Initialize `VotingClassifier` with
       `voting='soft'`.
    b. Ensure each SVM model can output
       probabilities (`probability=True`).
Train the Voting Classifier:
  - Fit the `VotingClassifier` on the
    training data.
Evaluate the ensemble model:
  - Predict on the test set using the
    Voting Classifier.
  - Calculate performance metrics
    (accuracy, precision, recall, F1-
    score).
Output the results:
  - Print the evaluation metrics for both
    hard and soft voting.
End.
```

2.5 Evaluation Model

Model performance was summarized with a confusion matrix, which records four outcomes: true positives (TP)—positive instances correctly identified as positive; true negatives (TN)—negative instances correctly identified as negative; false positives (FP)—negative instances mistakenly labeled as positive; and false negatives (FN)—positive instances incorrectly labeled as negative.

In addition to the performance indicators, evaluation metrics derived from the Confusion Matrix included in Equations 10 through 13.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (10)$$

$$Precision = \frac{TP}{TP+FP} \quad (11)$$

$$Recall = \frac{TP}{TP+FN} \quad (12)$$

$$F1-Score = 2 \times \frac{Precision \times Recall}{Precision+Recall} \quad (13)$$

The evaluation will compare the classification performance of each individual kernel (Linear, RBF, Polynomial, and Sigmoid) against the ensemble model using both hard voting and soft voting mechanisms. This comparison will help determine whether combining multiple kernels provides significant performance improvements over using a single kernel alone.

3. Results and Discussions

This section presents the results obtained from the implementation and evaluation of the proposed sentiment analysis model using SVM Porlis with ensemble soft voting. The discussion includes performance comparisons between individual SVM kernels and ensemble models, along with insights gained from applying the SMOTE technique to address class imbalance. Evaluation metrics such as accuracy, precision, recall, F1-score, and the confusion matrix are analyzed to determine the model's effectiveness and robustness. In addition, a comparative analysis with previous studies is presented to highlight the improvement achieved by the proposed approach.

3.1 Results

This research applied the Synthetic Minority Over-sampling Technique (SMOTE) to overcome class imbalance within the sentiment dataset. Class imbalance where one sentiment class (positive or negative) dominates the data can bias the model toward the majority class and result in poor generalization for minority sentiment. SMOTE addresses this by generating synthetic data points for the minority class through interpolation, resulting in a balanced dataset. After applying SMOTE, the dataset was trained using multiple SVM kernels: Linear, RBF, Polynomial, and Sigmoid, followed by ensemble learning using both soft and hard voting strategies.

Table 2. Classification Report using SVM Porlis Soft Voting

Class	Precision	Recall	F1-score	Support
0	0.98	0.98	0.98	599
1	0.98	0.98	0.98	583
accuracy			0.98	1182
macro avg	0.98	0.98	0.98	1182
weighted avg	0.98	0.98	0.98	1182

Table 2 presents the classification report for the SVM Porlis Soft Voting model, which combines predictions from all kernels through soft voting. Evaluation metrics such as precision, recall, and F1-score achieved 0.98 for both class 0 and class 1, indicating excellent classification performance. Out of 1,182 total samples, 98% were correctly predicted by the model, resulting in an overall accuracy of 0.98. Moreover, both macro and weighted averages reached 0.98, confirming that the model is not only accurate but also well-balanced across classes.

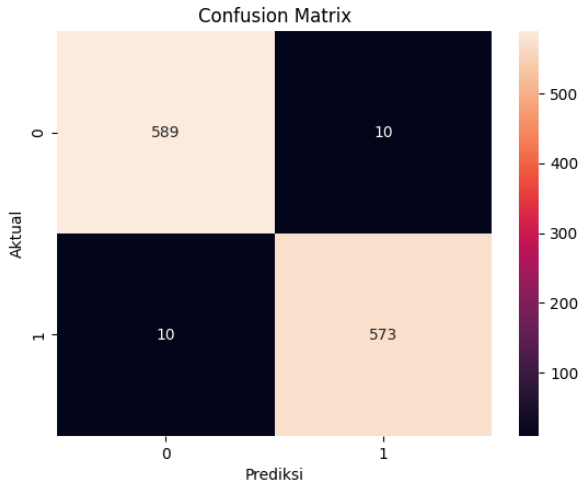


Figure 4. Confusion Matrix using SVM Porlis Soft Voting

The confusion matrix in Figure 4 offers a clear picture of the classifier's behavior. For class 0 (negative), 589 of 599 reviews were labeled correctly, with 10 false positives. For class 1 (positive), 573 of 583 reviews were correctly identified, with 10 false negatives. The equal counts of FP and FN indicate no class preference, while the very small error rates point to strong sensitivity and specificity. Overall, the balanced mistakes suggest the model generalizes well to unseen data, limiting both Type I (false positive) and Type II (false negative) errors in sentiment prediction.

3.2 Discussions

To evaluate the comparative effectiveness of the proposed approach, each individual SVM kernel was tested and compared against the ensemble variants. The results are presented in Table 3.

Table 3. Overall Testing Result

Algorithm	Accuracy	Precision	Recall	F1-Score
SVM Kernel RBF	93%	93%	93%	92%
SVM Kernel Linear	95%	95%	95%	95%
SVM Kernel Polynomial	51%	75%	51%	36%
SVM Kernel Sigmoid	95%	95%	95%	95%
SVM Porlis Soft Voting	98%	98%	98%	98%
SVM Porlis Hard Voting	96%	96%	96%	96%

As shown in Table 3, the RBF, Linear, and Sigmoid kernels demonstrate strong individual performance, with accuracy ranging from 93% to 95%. In contrast, the Polynomial kernel exhibits a significant performance drop, with accuracy falling to just 51% and an F1-score as low as 36%. This underperformance is likely due to the Polynomial kernel's sensitivity to noise and its tendency to overfit on complex, high-dimensional data such as informal tweet text. Without precise tuning of parameters—particularly the degree and constant coefficient (coef0)—this kernel can construct overly complex decision boundaries that fail to generalize across diverse language structures.

The proposed SVM Porlis Soft Voting model, however, outperformed all individual and ensemble models, achieving 98% across all performance metrics, including accuracy, precision, recall, and F1-score. This remarkable result is attributed to the soft voting mechanism, which integrates decision boundaries from four different kernels. Each kernel contributes its own strengths: RBF effectively captures non-linear sentiment patterns, Linear is well-suited for sparse text data, Sigmoid adapts to moderate non-linearities, and even the underperforming Polynomial kernel can capture multi-dimensional relationships within specific subsets of data.

Through soft voting, the model averages the probability predictions from all base classifiers rather than relying solely on majority class votes. This probabilistic aggregation improves decision stability, especially for tweets with ambiguous or borderline sentiment expressions.

Furthermore, the model's high and consistent performance across all metrics, along with its low misclassification rate, suggests minimal risk of overfitting. Its generalization ability is supported by balanced performance across both sentiment classes and reliable predictions, as observed in the confusion matrix.

In conclusion, the SVM Porlis Soft Voting model proves to be highly effective and generalizable for sentiment classification tasks, particularly when dealing with imbalanced and informal textual data such as Indonesian tweets. Its superior performance stems from the combination of diverse kernel perspectives, probabilistic decision aggregation, and well-balanced data preparation.

To ensure a consistent basis for comparison with previous studies, accuracy is selected as the primary evaluation metric in Table 3. Accuracy is a widely used performance measure in sentiment classification tasks due to its simplicity and interpretability. It reflects the proportion of correctly classified instances out of the total samples and provides a general overview of model effectiveness. While other metrics such as precision, recall, or F1-score are useful in specific contexts, especially with imbalanced datasets, many prior studies only report accuracy. Therefore, this metric is used here to maintain fairness and comparability across different approaches.

This study is also compared with several previous works to evaluate how well the SVM Porlis approach with ensemble voting improves classification performance. The comparison is presented in Table 4, which outlines the accuracy and algorithms used in earlier studies. By examining these results, it is evident that the proposed model demonstrates a significant

improvement in both accuracy and consistency compared to the methods employed in prior research.

Table 4. Comparison with Previous Research

Researchers	Algorithm	Accuracy
Hokijuliandy et al. 2023 [10]	SVM + Chi-Square	96%
Ramasamy et al. 2021 [42]	SVM + Nature-inspired Optimization	87%
Supriyatna & Putri, 2024 [43]	SVM + PSO	97%
Hidayat & Wibowo, 2024 [44]	SVM + Information Gain	89%
Imanuddin et al. 2023 [45]	SVM Kernel Linear	91%
Susanto & Suparwati, 2023 [46]	SVM + PSO	80%
Anam et al. 2022 [47]	SVM + Adaboost	92%
This Research	SVM Porlis Hard Voting	96%
	SVM Porlis Soft Voting	98%

Table 4 presents a comparison between the proposed method in this study namely the SVM Porlis Soft Voting and Hard Voting models and several previous studies that employed various combinations of SVM algorithms and performance enhancement techniques. The comparison shows that the SVM Porlis Soft Voting model achieved an accuracy of 98%, making it the most superior approach among the listed methods. This result outperforms the SVM + PSO approach by Supriyatna & Putri (2024), which achieved 97%, and the SVM + Chi-Square method by [10] with 96% accuracy. The model also significantly surpasses the SVM + Adaboost technique used by [47], which only reached 92%, and feature selection-based methods such as SVM + Information Gain by [44], which recorded 89% accuracy.

The SVM Linear Kernel approach tested by [45] achieved an accuracy of 91%, while another SVM + PSO method by [46] recorded only 80%, the lowest among the compared results. Compared to these outcomes, the SVM Porlis Soft Voting model demonstrated a +1% improvement over the previously best-performing method and up to +18% improvement over the lowest-performing method. The SVM Porlis Hard Voting model also delivered competitive performance with 96% accuracy, matching Hokijuliandy et al.'s results, though still slightly below the soft voting version.

This significant performance gain can be attributed to two key factors. First, the integration of multiple kernel types (Linear, RBF, Polynomial, and Sigmoid) enables the model to capture diverse patterns within the data—both linear and non-linear—more comprehensively. Second, the application of the soft voting strategy, which takes into account the prediction probabilities from each individual kernel before determining the final class, allows the model to be more adaptive to data ambiguity. This contrasts with hard voting, which relies solely on majority votes without considering each model's confidence level.

Furthermore, the inclusion of the SMOTE (Synthetic Minority Over-sampling Technique) method plays a crucial role in enhancing model performance by addressing the issue of class imbalance, which often causes bias toward the majority class. Therefore, the SVM Porlis Soft Voting approach excels not only in terms of accuracy but also demonstrates strong robustness in handling complex and imbalanced real-world datasets.

In conclusion, these results confirm that the multi-kernel ensemble model optimized with soft voting presents a highly promising solution for sentiment classification tasks and outperforms many existing SVM-based approaches reported in prior studies.

4. Conclusions

This study demonstrates the effectiveness of the SVM Porlis model, an ensemble approach that integrates multiple SVM kernels (Linear, RBF, Polynomial, and Sigmoid) using soft voting techniques. Unlike single-kernel models that rely on a singular decision boundary, this model leverages the strengths of each kernel to capture diverse sentiment patterns, ranging from linear relationships to complex non-linear structures. The use of soft voting, which considers prediction probabilities, represents a methodological advantage as it enables the model to make more accurate decisions, particularly in ambiguous cases. This multi-kernel integration is a unique approach that remains relatively unexplored in sentiment analysis of Indonesian-language social media data.

The model achieved an accuracy of 98%, significantly outperforming the individual kernel performances as well as previous SVM-based methods. This high level of performance was attained through a robust preprocessing pipeline and data balancing using SMOTE, which effectively addressed the issue of class imbalance. However, the polynomial kernel underperformed due to its tendency to overfit, highlighting the importance of specific parameter tuning for each kernel within an ensemble framework.

Despite the promising results, several limitations must be acknowledged. The model was tested only within a specific domain—tweets in Bahasa Indonesia related to the naturalization of football players—so its generalizability to other domains remains uncertain. In addition, the model has not yet been evaluated in real-time scenarios, where latency, data quality, and scalability are critical factors. Potential data bias also warrants attention, as social media opinions do not always reflect the broader population and may be influenced by trends, echo chambers, or bot activity.

For future research, several directions can be pursued. First, the model should be evaluated on different domains and data types to assess its generalizability beyond the context of football-related naturalization issues in Indonesia. Second, testing in real-time environments is necessary to measure the model's

performance in handling streaming data with low latency. Third, exploring more adaptive parameter optimization techniques such as Bayesian Optimization or Optuna could further enhance the performance of each kernel within the ensemble. Additionally, incorporating other ensemble strategies such as stacking or kernel-based boosting may offer alternatives to improve accuracy and classification stability. With these expansions, the SVM Porlis model can serve as a strong foundation for building more resilient and applicable sentiment classification systems across various social and linguistic contexts

Acknowledgements

The author reports no competing interests. This study received no dedicated funding from governmental, commercial, or nonprofit organizations.

References

[1] M. K. Anam, M. B. Firdaus, F. Suandi, Lathifah, T. Nasution, and S. Fadly, "Performance Improvement of Machine Learning Algorithm Using Ensemble Method on Text Mining," in *ICFTSS 2024 - International Conference on Future Technologies for Smart Society*, Kuala Lumpur: Institute of Electrical and Electronics Engineers Inc., Sep. 2024, pp. 90–95. doi: 10.1109/ICFTSS61109.2024.10691363.

[2] R. Guido, S. Ferrisi, D. Lofaro, and D. Conforti, "An Overview on the Advancements of Support Vector Machine Models in Healthcare Applications: A Review," *Information (Switzerland)*, vol. 15, no. 4, pp. 1–36, Apr. 2024, doi: 10.3390/info15040235.

[3] A. Zamsuri, S. Defit, and G. W. Nurcahyo, "Development and Comparison of Multiple Emotion Classification Models in Indonesia Text Using Machine Learning," *Journal of Advances in Information Technology*, vol. 15, no. 4, pp. 519–531, 2024, doi: 10.12720/jait.15.4.519-531.

[4] N. Amaya-Tejera, M. Gamarra, J. I. Vélez, and E. Zurek, "A distance-based kernel for classification via Support Vector Machines," *Front Artif Intell*, vol. 7, pp. 1–15, Feb. 2024, doi: 10.3389/frai.2024.1287875.

[5] J. Nalepa and M. Kawulok, "Selecting training sets for support vector machines: a review," *Artif Intell Rev*, vol. 52, no. 2, pp. 857–900, Aug. 2019, doi: 10.1007/s10462-017-9611-1.

[6] M. A. Sembiring, H. Saputra, R. A. Yusda, S. Sutarman, and E. B. Nababan, "Performance of Robust Support Vector Machine Classification Model on Balanced, Imbalanced and Outliers Datasets," *JITK (Jurnal Ilmu Pengetahuan dan Teknologi Komputer)*, vol. 10, no. 1, pp. 208–215, Aug. 2024, doi: 10.33480/jitk.v10i1.5272.

[7] W. Sholihah and A. Silvia Handayani, "Revolutionizing Healthcare: Comprehensive Evaluation and Optimization of SVM Kernels for Precise General Health Diagnosis," *Scientific Journal of Informatics*, vol. 10, no. 4, pp. 445–454, 2023, doi: 10.15294/sji.v10i4.46430.

[8] R. A. Sulthana, A. K. Jaithunbi, H. Harikrishnan, and V. Varadarajan, "Sentiment Analysis on Movie Reviews Dataset Using Support Vector Machines and Ensemble Learning," *International Journal of Information Technology and Web Engineering*, vol. 17, no. 1, pp. 1–23, 2022, doi: 10.4018/IJITWE.311428.

[9] M. Khalid, I. Ashraf, A. Mehmood, S. Ullah, M. Ahmad, and G. S. Choi, "GBSVM: Sentiment classification from unstructured reviews using ensemble classifier," *Applied Sciences (Switzerland)*, vol. 10, no. 8, Apr. 2020, doi: 10.3390/APP10082788.

[10] E. Hokijuliandy, H. Napitupulu, and Firdaniza, "Application of SVM and Chi-Square Feature Selection for Sentiment Analysis of Indonesia's National Health Insurance Mobile Application," *Mathematics*, vol. 11, no. 17, pp. 1–21, Sep. 2023, doi: 10.3390/math11173765.

[11] V. KP, R. AB, G. HL, V. Ravi, and M. Krichen, "A tweet sentiment classification approach using an ensemble classifier," *International Journal of Cognitive Computing in Engineering*, vol. 5, pp. 170–177, Jan. 2024, doi: 10.1016/j.ijcce.2024.04.001.

[12] M. Liebenlito, N. Inayah, E. Choerunnisa, T. E. Sutanto, and S. Inna, "Active Learning on Indonesian Twitter Sentiment Analysis Using Uncertainty Sampling," *Journal of Applied Data Sciences*, vol. 5, no. 1, pp. 114–121, Jan. 2024, doi: 10.47738/jads.v5i1.144.

[13] N. Mardiah, L. Marlina, K. Khairul, Z. Sitorus, and M. Iqbal, "Analysis Of Indonesian People's Sentiment Towards 2024 Presidential Candidates On Social Media Using Naïve Bayes Classifier and Support Vector Machine," *Building of Informatics, Technology and Science (BITS)*, vol. 6, no. 2, pp. 950–960, Sep. 2024, doi: 10.47065/bits.v6i2.5766.

[14] I. G. B. A. Budaya and I. K. P. Suniantara, "Comparison of Sentiment Analysis Algorithms with SMOTE Oversampling and TF-IDF Implementation on Google Reviews for Public Health Centers," *MALCOM: Indonesian Journal of Machine Learning and Computer Science*, vol. 4, no. 3, pp. 1077–1086, Jul. 2024, doi: 10.57152/malcom.v4i3.1459.

[15] N. Saha, A. K. Show, P. Das, and S. Nanda, "Performance comparison of different kernel tricks based on SVM approach for parkinson's disease detection," in *2021 2nd International Conference for Emerging Technology, INCET 2021*, Institute of Electrical and Electronics Engineers Inc., May 2021, pp. 1–4. doi: 10.1109/INCET51464.2021.9456233.

[16] X. Ding, J. Liu, F. Yang, and J. Cao, "Random radial basis function kernel-based support vector machine," *J Franklin Inst*, vol. 358, no. 18, pp. 10121–10140, Dec. 2021, doi: 10.1016/j.jfranklin.2021.10.005.

[17] S. D. Latif *et al.*, "Improving sea level prediction in coastal areas using machine learning techniques," *Ain Shams Engineering Journal*, vol. 15, no. 9, pp. 1–21, Sep. 2024, doi: 10.1016/j.asej.2024.102916.

[18] Z. Abidin, W. Destian, and R. Umer, "Combining support vector machine with radial basis function kernel and information gain for sentiment analysis of movie reviews," in *Journal of Physics: Conference Series*, IOP Publishing Ltd, Jun. 2021, pp. 1–5. doi: 10.1088/1742-6596/1918/4/042157.

[19] H. Prasetya, Z. Situmorang, and R. Rosnelly, "SVM Optimization with Kernel Function for Sentiment Analysis on Social Media twitter (X) in AFC U23 Asian Cup Case Study," in *1st Proceeding of International Conference on Science and Technology UISU (ICST)*, 2024, pp. 227–233. doi: 10.30743/wjxmmr59.

[20] A. F. Rochim, K. Widyaningrum, and D. Eridani, "Performance Comparison of Support Vector Machine Kernel Functions in Classifying COVID-19 Sentiment," in *International Seminar on Research of Information Technology and Intelligent Systems*, Institute of Electrical and Electronics Engineers Inc., 2021, pp. 224–228. doi: 10.1109/ISRITI54043.2021.9702845.

[21] F. M. Rizky, J. Jondri, and K. M. Lhaksmana, "Twitter Sentiment Analysis of Kanjuruhan Disaster using Word2Vec and Support Vector Machine," *Building of Informatics, Technology and Science (BITS)*, vol. 5, no. 1, pp. 219–227, Jun. 2023, doi: 10.47065/bits.v5i1.3612.

[22] I. S. Al-Mejibli, J. K. Alwan, and D. H. Abd, "The effect of gamma value on support vector machine performance with different kernels," *International Journal of Electrical and Computer Engineering*, vol. 10, no. 5, pp. 5497–5506, Oct. 2020, doi: 10.11591/IJECE.V10I5.PP5497-5506.

[23] N. K. M. Budayani, I. Slamet, and S. S. Handajani, "A Comparison of SVM Kernel Functions for Sentiment Analysis of UU TPKS," *Sci Educ (Dordr)*, vol. 2, pp. 761–765, 2023.

[24] C. B. Tan, M. H. A. Hijazi, and P. N. E. Nohuddin, "A comparison of different support vector machine kernels for artificial speech detection," *Telkomnika (Telecommunication Computing Electronics and Control)*, vol. 21, no. 1, pp. 97–103, Feb. 2023, doi: 10.12928/TELKOMNIKA.v21i1.24259.

[25] A. Nurkholis, D. Alita, and A. Munandar, "Comparison of Kernel Support Vector Machine Multi-Class in PPKM Sentiment Analysis on Twitter," *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 6, no. 2, pp. 227–233, Apr. 2022, doi: 10.29207/resti.v6i2.3906.

[26] D. Aryo Anggoro and D. Permatasari, "Performance Comparison of the Kernels of Support Vector Machine Algorithm for Diabetes Mellitus Classification," *Int J Adv Comput Sci Appl*, vol. 14, no. 2, p. 2023, 2023, doi: 10.14569/IJACSA.2023.0140226.

[27] M. A. Nanda, K. B. Seminar, D. Nandika, and A. Maddu, "A comparison study of kernel functions in the support vector machine and its application for termite detection," *Information (Switzerland)*, vol. 9, no. 1, pp. 1–14, Jan. 2018, doi: 10.3390/info9010005.

[28] M. K. Anam *et al.*, "Sara Detection on Social Media Using Deep Learning Algorithm Development," *Journal of Applied Engineering and Technological Science*, vol. 6, no. 1, pp. 225–237, Dec. 2024, doi: 10.37385/jaets.v6i1.5390.

[29] fikrimln16, "data-crawling-x-tweetharvest."

[30] M. K. Anam, S. Defit, Haviluddin, L. Efrizoni, and M. B. Firdaus, "Early Stopping on CNN-LSTM Development to Improve Classification Performance," *Journal of Applied Data Sciences*, vol. 5, no. 3, pp. 1175–1188, 2024, doi: 10.47738/jads.v5i3.312.

[31] F. Suandi *et al.*, "Enhancing Sentiment Analysis Performance Using SMOTE and Majority Voting in Machine Learning Algorithms," in *International Conference on Applied Engineering*, Atlantis Press, 2024, pp. 126–138. doi: 10.2991/978-94-6463-620-8_10.

[32] Hamdani, Randi N.A, and M. K. Anam, "Comparison of Support Vector Machine and Random Forest Algorithms for Analyzing Online Loans on Twitter social media," *JAIA-Journal Of Artificial Intelligence And Applications*, vol. 4, no. 1, pp. 8–16, 2024, doi: 10.33372/jaia.v4i1.1087.

[33] A. N. Ulfah, M. K. Anam, N. Y. S. Munti, S. Yaakub, and M. B. Firdaus, "Sentiment Analysis of the Convict Assimilation Program on Handling Covid-19," *JUITA: Jurnal Informatika*, vol. 10, no. 2, pp. 209–216, 2022, doi: 10.30595/juita.v10i2.12308.

[34] P. P. Putra, M. K. Anam, S. Defit, and A. Yunianta, "Enhancing the Decision Tree Algorithm to Improve Performance Across Various Datasets," *INTENSIF: Jurnal Ilmiah Penelitian dan Penerapan Teknologi Sistem Informasi*, vol. 8, no. 2, pp. 200–212, Aug. 2024, doi: 10.29407/intensif.v8i2.22280.

[35] M. K. Anam *et al.*, "Enhancing the Performance of Machine Learning Algorithm for Intent Sentiment Analysis on Village Fund Topic," *Journal of Applied Data Sciences*, vol. 6, no. 2, pp. 1102–1115, 2025, doi: 10.47738/jads.v6i2.637.

[36] M. K. Anam *et al.*, "Improved Performance of Hybrid GRU-BiLSTM for Detection Emotion on Twitter Dataset," *Journal of Applied Data Sciences*, vol. 6, no. 1, pp. 354–365, Jan. 2025, doi: 10.47738/jads.v6i1.459.

[37] M. K. Anam, T. P. Lestari, H. Yenni, T. Nasution, and M. B. Firdaus, "Enhancement of Machine Learning Algorithm in Fine-grained Sentiment Analysis Using the Ensemble," *ECTI Transactions on Computer and Information Technology (ECTI-CIT)*, vol. 19, no. 2, pp. 159–167, Mar. 2025, doi: 10.37936/ecti-cit.2025192.257815.

[38] M. K. Anam, T. A. Fitri, Agustin, Lusiana, M. B. Firdaus, and A. T. Nurhuda, "Sentiment Analysis for Online Learning using The Lexicon-Based Method and The Support Vector Machine Algorithm," *ILKOM Jurnal Ilmiah*, vol. 15, no. 2, pp. 290–302, 2023, doi: 10.33096/ilkom.v15i2.1590.290-302.

[39] V. V., R. A. C, R. Mohammed, S. K. V, and P. S. Kumthekar, "Support Vector Machine Implementation to Separate Linear and Non-Linear Dataset," *Saudi Journal of Engineering and Technology*, vol. 8, no. 1, pp. 4–15, Jan. 2023, doi: 10.36348/sjet.2023.v08i01.002.

[40] A. P. Gopi, R. N. S. Jyothi, V. L. Narayana, and K. S. Sandeep, "Classification of tweets data based on polarity using improved RBF kernel of SVM," *International Journal of Information Technology (Singapore)*, vol. 15, no. 2, pp. 965–980, Feb. 2023, doi: 10.1007/s41870-019-00409-4.

[41] L. Muflikhah, D. Joko Haryanto, A. Andy Soebroto, and E. Santoso, "High Performance of Polynomial Kernel at SVM Algorithm for Sentiment Analysis," *Journal of Information Technology and Computer Science*, vol. 3, no. 2, pp. 194–201, 2018, doi: 10.25126/jitecs.20183260.

[42] L. K. Ramasamy, S. Kadry, Y. Nam, and M. N. Meqdad, "Performance analysis of sentiments in Twitter dataset using SVM models," *International Journal of Electrical and Computer Engineering*, vol. 11, no. 3, pp. 2275–2284, Jun. 2021, doi: 10.11591/ijece.v11i3.pp2275-2284.

[43] B. L. Supriyatna and F. P. Putri, "Optimized support vector machine for sentiment analysis of game reviews," *International Journal of Informatics and Communication Technology (IJ-ICT)*, vol. 13, no. 3, p. 344, Dec. 2024, doi: 10.11591/ijict.v13i3.pp344-353.

[44] M. Hidayat and A. Wibowo, "SVM Optimization With Information Gain Feature Selection to Increase the Accuracy of Sentiment Analysis of Increasing The Cost of the Hajj," *Jurnal Teknik Informatika (Jutif)*, vol. 5, no. 4, pp. 579–591, Aug. 2024, doi: 10.52436/1.jutif.2024.5.4.2217.

[45] Shahmirul Hafizullah Imanuddin, Kusworo Adi, and Rahmat Gernowo, "Sentiment Analysis on Satusehat Application Using Support Vector Machine Method," *Journal of Electronics, Electromedical Engineering, and Medical Informatics*, vol. 5, no. 3, pp. 143–149, Jul. 2023, doi: 10.35882/jeemi.v5i3.304.

[46] N. W. Susanto and H. Suparwito, "SVM-PSO Algorithm for Tweet Sentiment Analysis #BesokSenin," *Indonesian Journal of Information Systems (IJIS)*, vol. 6, no. 1, pp. 36–47, 2023, doi: 10.24002/ijis.v6i1.7551.

[47] M. K. Anam, M. I. Mahendra, W. Agustin, Rahmaddeni, and Nurjayadi, "Framework for Analyzing Netizen Opinions on BPJS Using Sentiment Analysis and Social Network Analysis (SNA)," *Intensif*, vol. 6, no. 1, pp. 2549–6824, 2022, doi: 10.29407/intensif.v6i1.15870.