Comparative Analysis Between Information Gain and Particle Swarm Optimization in the Support Vector Machine Algorithm for Sentiment Analysis

Aisyah Larasati1,a), Shidqi Zakivarian Hanifarchie1,b), and Agus Rachmad Purnama2,c)

1Department of Mechanical and Industrial Engineering, Universitas Negeri Malang, Malang, Indonesia

2Department of Industrial Engineering, Universitas Nahdlatul Ulama, Sidoarjo, Indonesia

a) Corresponding author: [aisyah.larasati.ft@um.ac.id](mailto:aisyah.larasati.ft@um.ac.id)

b) [shidqi.zakivarian.2005166@students.um.ac.id](mailto:shidqi.zakivarian.2005166@students.um.ac.id)

c) [agus\_purnama.tin@unusida.ac.id](mailto:agus_purnama.tin@unusida.ac.id)

**Abstract.**  The rapid technological advancements in the Industry 4.0 era have transformed various aspects of life, including the way we conduct transactions, with many now being done online through e-commerce. Understanding user opinions is crucial for improving services and maintaining customer loyalty. Sentiment analysis, which classifies opinions into positive or negative categories, is one method to achieve this. Using the Support Vector Machine (SVM) algorithm with appropriate feature selection techniques like Information Gain and Particle Swarm Optimization (PSO) can enhance model accuracy. This study uses 3,399 Tokopedia user reviews from the Google Playstore collected between January and May 2024, testing the SVM algorithm integrated with Information Gain and PSO. Results show that an SVM model without feature selection has an accuracy of 81.02%, with precision of 78%, recall of 86%, and F1-score of 82%. With Information Gain feature selection, accuracy increases to 84.26%, precision to 87%, recall to 82%, and F1-score to 84%. Finally, the SVM model with PSO feature selection achieves the highest accuracy of 84.56%, with precision of 81%, recall of 89%, and F1-score of 85%. Both Information Gain and PSO feature selection techniques improve the SVM model's performance, with PSO providing the highest accuracy improvement.

**Keywords:** Information Gain, Particle Swarm Optimization, Support Vector Machine, Sentiment Analysis

# INTRODUCTION

Sentiment analysis is a part of Natural Language Processing (NLP) focused on developing systems to detect and extract opinions from text [1]. In sentiment analysis, machine learning algorithms are applied to classify and understand the emotions or opinions contained in text. These algorithms aim to identify whether a text has a positive or negative tone by examining patterns within large datasets of text. Some machine learning algorithms used in sentiment analysis include Naive Bayes, Logistic Regression, Perceptron, K-Nearest Neighbor (KNN), and Support Vector Machine (SVM). Several studies have investigated sentiment analysis using machine learning algorithms, such as research by Putri and Kharisudin, which compared SVM, Naive Bayes, and Logistic Regression algorithms [2]. The comparison results showed that the SVM algorithm had the best performance with an accuracy rate of 97%. Subsequent research by Nasution and Hayaty compared the performance of SVM with K-Nearest Neighbor in sentiment analysis [3]. The findings indicated that the SVM algorithm achieved the highest classification accuracy of 89%. These studies suggest that SVM generally offers higher accuracy compared to other algorithms. Therefore, this study will use the SVM algorithm to classify and understand emotions or opinions in text for sentiment analysis.

Support Vector Machine (SVM) is a supervised learning algorithm that can analyze data and recognize classification patterns. SVM uses techniques to find the best hyperplane that separates data observations into different target classes. This hyperplane can be a line in two dimensions or a flat plane in multiple dimensions. SVM is a top-performing machine learning algorithm that handles high-dimensional data but has limitations in processing large datasets. The performance of a classification algorithm is influenced by several factors, including feature selection in sentiment analysis. A common issue in text-based sentiment analysis classification is the large number of features, which can lead to high data volume and suboptimal model accuracy. Feature selection is crucial for optimizing model performance by eliminating less relevant features. However, features must be carefully selected to avoid overfitting and underfitting. Information Gain is one method used for feature selection.

Information Gain is a method used in machine learning to determine how informative a feature is in distinguishing between positive and negative opinions. Information Gain is widely used in sentiment analysis because of its ability to enhance model performance by selecting the most relevant features. According to research by Ate and Nuraminah, applying Information Gain to the SVM model in movie review classification resulted in the highest accuracy model of 81%, compared to other feature selection methods like Chi Square, Forward Selection, and Backward Elimination [4]. Another study by Maulana et al. also showed that using Information Gain in the SVM model improved accuracy from 83% to 85% [5]. These findings affirm that Information Gain is effective not only in selecting important features but also in enhancing overall model performance, especially when applied to the SVM algorithm in sentiment analysis.

Another feature selection method is Particle Swarm Optimization (PSO). PSO is an optimization algorithm inspired by the behavior of bird flocks searching for food. Feature selection using PSO helps find the best solution in a complex search space, potentially leading to better feature subsets and optimizing overall analysis performance. PSO has several advantages, such as fewer parameters, ease of implementation, fast convergence, and simplicity, making it commonly used in function optimization, conventional optimization methods, and pattern classification. According to research by Pranomo et al., applying PSO in sentiment classification related to Google Classroom using the Naive Bayes algorithm improved model accuracy by up to 9.9%, making it the best feature selection method compared to N-gram and Information Gain [6]. Additionally, research by Handayani also demonstrated that PSO could improve SVM model accuracy by 5%, confirming PSO's effectiveness in optimizing machine learning algorithms for sentiment analysis [7].

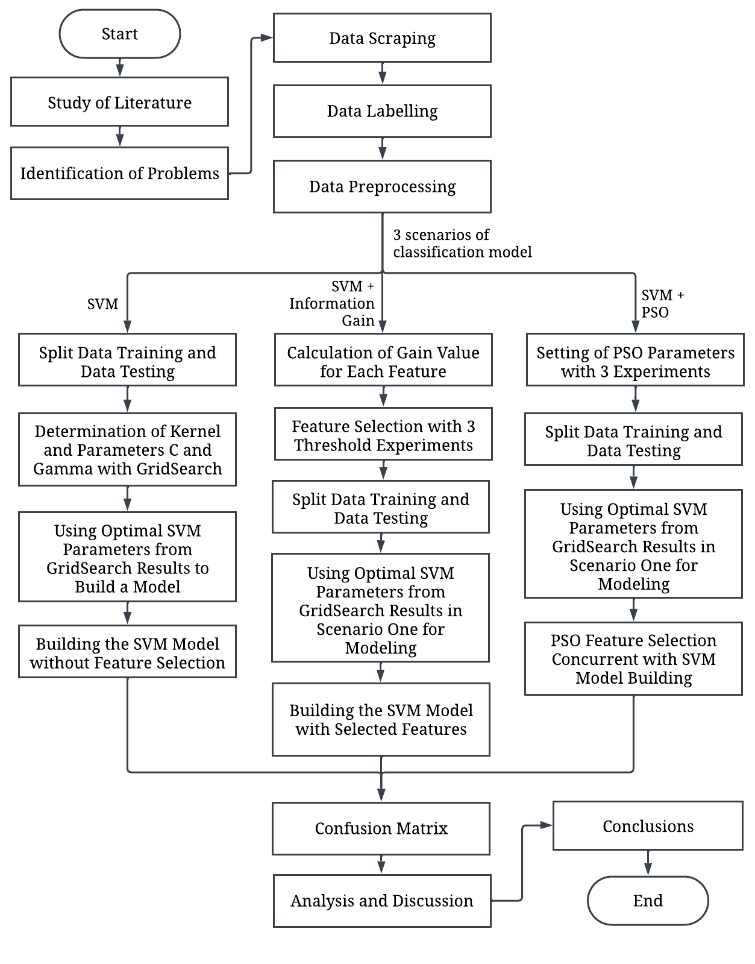
To implement sentiment analysis with the algorithms and methods described above, an object or case study is needed to assess opinion trends on the topic. One interesting phenomenon to study is the changes in buying and selling transactions in society. In today's Industry 4.0 era, rapid technological advancements have transformed various aspects of life. A noticeable change is the shift from traditional buying and selling to online transactions or e-commerce. The growth of e-commerce has led to many marketplaces emerging in Indonesia. A popular marketplace is Tokopedia, which has been downloaded over 100 million times. This large user base generates numerous reviews. To understand this, a method to classify user reviews into two categories, positive and negative, is required. One method to determine the trend of Tokopedia app user reviews is sentiment analysis.

Based on these issues, the authors propose a study titled "Comparative Analysis Between Information Gain and Particle Swarm Optimization in the Support Vector Machine Algorithm for Sentiment Analysis." This study applies the SVM algorithm to analyze sentiments in Tokopedia app reviews on the Google Playstore, integrating Information Gain and PSO feature selection to optimize the model. The results can help Tokopedia management understand user opinions and trends, as well as serve as a reference for researchers conducting similar studies with SVM, Information Gain, and PSO for sentiment analysis.

The study aims to achieve four objectives: first, determine the classification results of the SVM algorithm without feature selection for sentiment analysis of Tokopedia app reviews. Second, determine the results of the SVM model using Information Gain feature selection for sentiment analysis. Third, determine the results of the SVM model using PSO feature selection for sentiment analysis. And fourth, compare the SVM model results without feature selection, with Information Gain feature selection, and with PSO feature selection for sentiment analysis of Tokopedia app reviews.

# METHODS

This research is a quantitative experimental study that applies various techniques and draws conclusions at the end. Experimental research aims to determine the effects of input variable parameters set by the researcher. This study aims to compare the performance of the Support Vector Machine (SVM) algorithm with Information Gain and Particle Swarm Optimization (PSO) feature selection in sentiment analysis classification. The results will help formulate the best classification model to understand public responses to the Tokopedia app. To achieve the desired results, a clear and structured research method is essential. The complete research stages are shown in **FIGURE 1**.



**FIGURE 1.** Research Stages

The first stage involves identifying the existing problem: user complaints or opinions about the Tokopedia app expressed in reviews on the Google Playstore. The next stage is data collection in the form of Tokopedia app user reviews using web scraping to extract necessary data from the Google Playstore website. Extracted data include usernames, review texts, upload dates, and ratings. Scraped data are saved in CSV format to facilitate preprocessing.

The subsequent stage involves labeling the sentiment classes (positive or negative) of the collected data. In this study, positive labels are assigned to reviews with ratings of four and five, while negative labels are assigned to reviews with ratings of one, two, and three. After labeling, the next stage is text preprocessing, including case folding (converting all text to lowercase), data cleaning (removing characters, numbers, symbols, and emojis), normalization (converting non-standard words to standard ones), stemming (reducing words to their base forms), stopword removal (removing irrelevant words), tokenizing (splitting text into individual tokens or words), and feature extraction (converting text to numerical vector representations).

After preprocessing, the next step is building and classifying the model using Support Vector Machine (SVM). This process involves three scenarios: without feature selection, with Information Gain feature selection, and with Particle Swarm Optimization (PSO) feature selection. Each scenario involves splitting the data into training and testing sets to build the model. In the first scenario, GridSearchCV is used to determine the optimal C (complexity) and gamma parameters for SVM. In the second and third scenarios, experiments will determine the optimal parameter values for each feature selection method. In scenario two, experiments will test different thresholds to select features for building the model. Thresholds tested include 25%, 50%, and 75%. In scenario three, experiments will test PSO parameter combinations to find the best-performing model. Parameters tested include c1, c2, w, max iteration, particle number, and threshold, as shown in **TABLE 1**.

**TABLE 1.** PSO Parameter Testing Experiment

|  |  |  |
| --- | --- | --- |
| **Experiment** | **Parameter** | **Value** |
| 1 | c1, c2, and w on the value of the maximum number of iterations | 25 |
| 50 |
| 75 |
| 100 |
| 2 | Number of particles | 2 |
| 4 |
| 6 |
| 8 |
| 3 | Threshold | 0.1 |
| 0.3 |
| 0.5 |
| 0.7 |

The next stage involves evaluating the SVM classification model for each scenario based on accuracy. Accuracy is a common and effective performance metric for balanced and imbalanced data [8]. Accuracy can be derived from the Confusion Matrix, which also provides precision (the accuracy of actual data with predicted results), recall (the model's success rate in finding information), and F1-score (the harmonic mean of recall and precision). The final stage is analyzing the comparative results of the SVM classification model after it has been built. The analysis involves comparing the accuracy of the SVM model without feature selection, the SVM model with Information Gain feature selection, and the SVM model with PSO feature selection. The model with the highest accuracy will be identified as the best model, followed by drawing conclusions from the overall research results.

# RESULT AND DISCUSSION

## OVERVIEW OF RESEARCH DATA

In this study, sentiment analysis was conducted using the Support Vector Machine (SVM) algorithm and divided into three model-building scenarios. The first model was built without feature selection, the second with Information Gain feature selection, and the third with Particle Swarm Optimization (PSO) feature selection. The main stages of the sentiment analysis included data collection, sentiment class labeling, text preprocessing, model building with three scenarios, and model evaluation using the Confusion Matrix. Data collection from January to May 2024 on the Google Playstore yielded 3,399 reviews, as shown in **FIGURE 2**.

A graph of reviews based on ratings

Description automatically generated

**FIGURE 2.** Number of Reviews Based on Ratings

The collected data were labeled with sentiment classes based on ratings, as shown in **FIGURE 3**. Positive labels were assigned to reviews with ratings of 4 and 5, while negative labels were assigned to reviews with ratings of 1, 2, and 3, based on previous research indicating that neutral ratings (3) tend to contain negative content [9]. The reviews underwent text preprocessing, including case folding, data cleaning, text normalization, stemming, stopword removal, tokenizing, and feature extraction. The final result was numerical data ready for model building, as shown in **TABLE 2**.

A graph of reviews based on negative

Description automatically generated

**FIGURE 3.** Number of Reviews Based on Sentiment

**TABLE 2.** Results of the Feature Extraction Process

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Reviews** | **application** | **goods** | **shopping** | **excellent** | **complete** | **...** | **zuhur** |
| **1** | 1 | 0 | 1 | 0 | 0 | ... | 0 |
| **2** | 0 | 0 | 0 | 1 | 0 | ... | 0 |
| **3** | 0 | 0 | 0 | 0 | 0 | ... | 0 |
| **4** | 1 | 0 | 0 | 1 | 1 | ... | 0 |
| **...** | ... | ... | ... | ... | ... | ... | ... |
| **3399** | 0 | 0 | 0 | 0 | 0 | ... | 0 |

## SUPPORT VECTOR MACHINE MODEL WITHOUT FEATURE SELECTION

Scenario 1 involves building a model without feature selection, including steps such as dividing the training and testing data, determining SVM parameters, and building the SVM model. The data is divided using an 80% training and 20% testing ratio. This ratio is based on previous research stating that an 80:20 ratio can provide better classification model accuracy and avoid overfitting or underfitting [10] [11]. Next, SVM parameters are determined using the GridSearchCV method on the RBF kernel. The choice of the RBF kernel is based on previous research indicating that the RBF kernel is the most optimal and provides the highest accuracy for SVM models [12]. Parameters tested using GridSearchCV include C and gamma, with values 0.01, 0.1, 1, 10, and 100. These values cover a wide range, making it easier to find the optimal parameters. The test results for each combination are shown in **TABLE 3**.

**TABLE 3.** SVM Parameter Test Results

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **No** | ***Complexity*** | ***Gamma*** | **Accuracy** | **No** | ***Complexity*** | ***Gamma*** | **Accuracy** |
| **1** | 0.01 | 0.01 | 0.514158 | **14** | 1 | 10 | 0.534383 |
| **2** | 0.01 | 0.1 | 0.774185 | **15** | 1 | 100 | 0.534383 |
| **3** | 0.01 | 1 | 0.507171 | **16** | 10 | 0.01 | 0.818326 |
| **4** | 0.01 | 10 | 0.507171 | **17** | 10 | 0.1 | 0.822364 |
| **5** | 0.01 | 100 | 0.507171 | **18** | 10 | 1 | 0.687015 |
| **6** | 0.1 | 0.01 | 0.757632 | **19** | 10 | 10 | 0.534383 |
| **7** | 0.1 | 0.1 | 0.796617 | **20** | 10 | 100 | 0.534383 |
| **8** | 0.1 | 1 | 0.562333 | **21** | 100 | 0.01 | 0.803241 |
| **9** | 0.1 | 10 | 0.507171 | **22** | 100 | 0.1 | 0.820160 |
| **10** | 0.1 | 100 | 0.507171 | **23** | 100 | 1 | 0.687015 |
| **11** | 1 | 0.01 | 0.814279 | **24** | 100 | 10 | 0.534383 |
| **12** | 1 | 0.1 | 0.824572 | **25** | 100 | 100 | 0.534383 |
| **13** | 1 | 1 | 0.677451 |  |  |  |  |

**TABLE 4**. Confusion Matrix of SVM Model without Feature Selection

|  |  |  |  |
| --- | --- | --- | --- |
|  | | **Actual** | |
| **Positive** | **Negative** |
| **Prediction** | **Positive** | 286 | 48 |
| **Negative** | 81 | 265 |

Based on the GridSearchCV test, the parameter combination that provides the best accuracy is C=1 and gamma=0.1. The C parameter is used to adjust the model's margin sensitivity, where a small C value decreases the margin sensitivity, and conversely, a large C value increases the margin sensitivity [13]. Meanwhile, the gamma parameter is used to adjust the complexity of the model in data classification. A small gamma value tends to make the hyperplane linear and reduces the model's ability to capture data complexity, whereas a large gamma value makes the radius of the support vector area more complex. The next step is building the SVM model with the previously determined parameters on the RBF kernel with an 80:20 training and testing data ratio. According to **TABLE 4**, the SVM model built in Scenario 1 correctly predicts 286 positive cases and 265 negative cases but also results in 48 false positives and 81 false negatives. The evaluation metrics calculated from these results show an accuracy of 81.02%, precision of 78%, recall of 86%, and F1-score of 82%. This accuracy indicates that the Scenario 1 model is categorized as having good classification.

## SUPPORT VECTOR MACHINE MODEL WITH INFORMATION GAIN

In Scenario 2, a model applying Information Gain feature selection is built, including steps such as calculating Gain values, selecting features, dividing the dataset, and building the SVM model with the selected features. Gain values are calculated for all 4442 features in the dataset. Gain measures how much information a feature provides in separating data classes, involving entropy calculation before and after data separation based on the feature. After calculation, features are ranked by Gain values, as shown in **TABLE 5**, displaying the top 10 features.

**TABLE 5**. Gain Value Calculation Results for Features

|  |  |
| --- | --- |
| **Features** | **Gain Value** |
| disappointed | 0.038116 |
| telkomsel | 0.032068 |
| jabodetabek | 0.031942 |
| finished | 0.031792 |
| recommendation | 0.031575 |
| loss | 0.030637 |
| smule | 0.029736 |
| shopee | 0.029308 |
| sell | 0.029056 |
| bad | 0.028851 |

A graph of blue rectangular bars with numbers

Description automatically generated

**FIGURE 4**. Accuracy of Each Threshold Information Gain

Next, feature selection is based on three threshold experiments: 25%, 50%, and 75%. The threshold value used in this experiment refers to [14], who also conducted sentiment analysis using Information Gain feature selection. The tests show that using a 75% threshold for feature selection provides the best accuracy compared to others, as shown in **FIGURE 4**. As the threshold increases, the number of selected features also increases, from 1111 features at 25% to 3331 features at 75%. Additionally, model accuracy gradually improves, from 81% at 25%, to 83% at 50%, and reaching 84% at 75%. This indicates that increasing the threshold provides more relevant information, enhancing performance in sentiment analysis classification. However, at an 80% threshold, model accuracy drops to 82%. This decrease suggests that a too-high threshold may select too many less relevant features, reducing model efficiency and accuracy. Thus, choosing the right threshold is crucial to balance model complexity and performance.

After feature selection, the next step is building the model with previously determined SVM parameters, namely C=1 and gamma=0.1 on the RBF kernel with an 80:20 data split ratio. According to **TABLE 6**, the Scenario 2 SVM model correctly predicts 290 positive cases and 283 negative cases, with 44 false positives and 63 false negatives. The evaluation shows that the SVM model with Information Gain feature selection has an accuracy of 84.26%, precision of 87%, recall of 82%, and F1-score of 84%, categorized as good classification. The classification model in scenario 2 demonstrates that using Information Gain for feature selection can improve model accuracy. This is in line with [15], who used Information Gain on the Naive Bayes algorithm for sentiment analysis and found that Naive Bayes classification with feature selection resulted in better performance with a 7% accuracy increase.

**TABLE 6**. Confusion Matrix of SVM Model with Information Gain Threshold of 75%

|  |  |  |  |
| --- | --- | --- | --- |
|  | | **Actual** | |
| **Positive** | **Positive** |
| **Prediction** | **Positive** | 290 | 44 |
| **Negative** | 63 | 283 |

## SUPPORT VECTOR MACHINE MODEL WITH PARTICLE SWARM OPTIMIZATION

Scenario 3 involves building a model using Particle Swarm Optimization (PSO) feature selection, including steps such as determining PSO parameters, dividing the dataset, and selecting features while building the SVM model. Initially, PSO parameters are determined through three experiments: Experiment 1 on parameters c1, c2, w, and maximum iterations; Experiment 2 on the number of particles; and Experiment 3 on threshold value. The values used to test the PSO parameter in **TABLE 1** refer to [16], both of whom also used PSO feature selection for sentiment analysis. Parameter testing in Experiment 1 uses the skopt library, resulting in the best accuracy with c1=1.6, c2=0.4, w=0.8, and maximum iterations=75, achieving 84% accuracy.

After finding the optimal parameters in Experiment 1, Experiments 2 and 3 determine the number of particles and threshold values that provide the best accuracy. Unlike before, Experiments 2 and 3 require model building to determine accuracy. Models are built with an 80:20 data split ratio and Experiment 1 parameters, with each combination of parameters tested in Experiments 2 and 3. The best combination is shown in **TABLE** **7**, with 6 particles and a 0.1 threshold, selecting 4044 features and achieving 84.56% accuracy.

**TABLE 7**. Accuracy of Each Combination of Number of Particles and Threshold

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **No** | **Number of Particles** | **Threshold** | **Number of Feature** | **Accuracy** | **No** | **Number of Particles** | **Threshold** | **Number of Feature** | **Accuracy** |
| **1** | 2 | 0.1 | 3977 | 0.8441 | **9** | 6 | 0.1 | 4044 | 0.8456 |
| **2** | 2 | 0.3 | 3080 | 0.8221 | **10** | 6 | 0.3 | 3075 | 0.8221 |
| **3** | 2 | 0.5 | 2222 | 0.8206 | **11** | 6 | 0.5 | 2208 | 0.8000 |
| **4** | 2 | 0.7 | 1324 | 0.7956 | **12** | 6 | 0.7 | 1419 | 0.7779 |
| **5** | 4 | 0.1 | 3988 | 0.8353 | **13** | 8 | 0.1 | 4009 | 0.8368 |
| **6** | 4 | 0.3 | 3163 | 0.8338 | **14** | 8 | 0.3 | 3102 | 0.8235 |
| **7** | 4 | 0.5 | 2227 | 0.8132 | **15** | 8 | 0.5 | 2225 | 0.8118 |
| **8** | 4 | 0.7 | 1391 | 0.7912 | **16** | 8 | 0.7 | 1325 | 0.7691 |

The best-performing model, with c1=1.6, c2=0.4, w=0.8, maximum iterations=75, 6 particles, and a 0.1 threshold, is evaluated using a Confusion Matrix. According to **TABLE 8**, the model correctly predicts 298 positive cases and 277 negative cases, with 36 false positives and 69 false negatives. Additionally, the SVM model with Particle Swarm Optimization feature selection is categorized as good classification, with an accuracy of 84.56%, precision of 81%, recall of 89%, and F1-score of 85%. Overall, the model in scenario 3 shows that using PSO for feature selection can improve model accuracy. This is consistent with [17], who also applied PSO feature selection in sentiment analysis classification and found a 6% accuracy increase in the Naive Bayes algorithm model.

**TABLE 8**. Confusion Matrix of SVM Model with PSO

|  |  |  |  |
| --- | --- | --- | --- |
|  | | **Actual** | |
| **Positive** | **Positive** |
| **Prediction** | **Positive** | 298 | 36 |
| **Negative** | 69 | 277 |

## COMPARISON OF CLASSIFICATION RESULTS BETWEEN MODELS

A graph with blue rectangular bars

Description automatically generated

**FIGURE 5**. Model Accuracy for Each Scenario

Research results show that the SVM model without feature selection achieves 81.02% accuracy, indicating good performance, although not as optimal as when feature selection is applied, as shown in **FIGURE 5**. Combining SVM with Information Gain increases accuracy by 3.24% to 84.26%. Information Gain enhances SVM accuracy by selecting the most relevant features, calculating the information each feature contributes to data class separation. By selecting features with the highest Gain values, using a 75% threshold, the model focuses on attributes significantly contributing to classification. Additionally, Information Gain feature selection reduces noise and overfitting risk, making the model more efficient and accurate.

Combining SVM with Particle Swarm Optimization (PSO) increases accuracy by 3.54% to 84.56%. PSO optimizes SVM parameters, such as C and gamma. Through iterations, PSO particles adjust their positions based on individual and swarm experiences, finding parameter combinations that yield the best performance. Optimal parameter settings allow the SVM model to maximize the margin between classes, enhancing overall prediction accuracy. Overall, combining SVM with PSO achieves the highest accuracy, showing that this optimization technique is more effective in improving model performance compared to no feature selection or using Information Gain feature selection. This is in line with [6], which compared the performance of N-gram, Information Gain, and PSO feature selections and found that the most optimal model used PSO with a 9.9% accuracy increase.

# CONCLUSIONS

The study compares the performance of Support Vector Machine (SVM) models for classifying Tokopedia app reviews, both with and without feature selection methods. The SVM model without feature selection achieved an accuracy of 81.02%, precision of 78%, recall of 86%, and an F1-score of 82%, indicating good classification performance. When Information Gain was used as a feature selection method, the SVM model's performance improved, with accuracy increasing to 84.26%, precision to 87%, recall to 82%, and the F1-score to 84%.

Further improvement was observed with the Particle Swarm Optimization (PSO) feature selection, where the SVM model achieved the highest accuracy of 84.56%, precision of 81%, recall of 89%, and an F1-score of 85%. The results suggest that both Information Gain and PSO feature selection methods enhance the performance of SVM models in sentiment analysis of Tokopedia app reviews, with PSO proving to be the most effective technique, leading to the highest accuracy among the models tested.

# REFERENCES

1. Arsi, P., R. Wahyudi, and R. Waluyo, *Optimasi SVM Berbasis PSO pada Analisis Sentimen Wacana Pindah Ibu Kota Indonesia.* Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi), 2021. **5**(2): p. 231-237.

2. Putri, M.I. and I. Kharisudin. *Analisis Sentimen Pengguna Aplikasi Marketplace Tokopedia Pada Situs Google Play Menggunakan Metode Support Vector Machine (SVM), Naïve Bayes, dan Logistic Regression*. in *PRISMA, Prosiding Seminar Nasional Matematika*. 2022.

3. Nasution, M.R.A. and M. Hayaty, *Perbandingan akurasi dan waktu proses algoritma K-NN dan SVM dalam analisis sentimen twitter.* J. Inform, 2019. **6**(2): p. 226-235.

4. Ate, I.S. and A. Nuraminah, *Komparasi Algoritma Feature Selection Pada Analisis Sentimen Review Film.* Jurnal Ilmiah Teknik Informatika dan Komunikasi, 2022. **2**(2): p. 96-102.

5. Maulana, R., et al., *Improved Accuracy of Sentiment Analysis Movie Review Using Support Vector Machine Based Information Gain.* Journal of Physics: Conference Series, 2020. **1641**(1): p. 012060.

6. Pramono, F., D. Rosiyadi, and W. Gata, *Integrasi N-gram, Information Gain, Particle Swarm Optimation di Naïve Bayes untuk Optimasi Sentimen Google Classroom.* Jurnal RESTI (Rekayasa Sistem Dan Teknologi Informasi), 2019. **3**(3): p. 383-388.

7. Handayani, R.N., *Optimasi Algoritma Support Vector Machine untuk Analisis Sentimen pada Ulasan Produk Tokopedia Menggunakan PSO.* Media Informatika, 2021. **20**(2): p. 97-108.

8. Tharwat, A., *Classification assessment methods.* Applied computing and informatics, 2021. **17**(1): p. 168-192.

9. Jagdale, R.S., V.S. Shirsat, and S.N. Deshmukh. *Sentiment Analysis on Product Reviews Using Machine Learning Techniques*. in *Cognitive Informatics and Soft Computing*. 2019. Singapore: Springer Singapore.

10. Istiadi, I. and R. Aviv Yuniar, *OPTIMISASI PARAMETER SUPPORT VECTOR MACHINE BERBASIS ALGORITMA GENETIKA PADA KLASIFIKASI TEKS PENGADUAN MASYARAKAT.* Conference on Innovation and Application of Science and Technology (CIASTECH), 2020(CIASTECH 2020 "Peranan Strategis Teknologi Dalam Kehidupan di Era New Normal"): p. 481-488.

11. Rahman, M. and A. Pambudi, *Identifikasi Citra Daun Selada Dalam Menentukan Kualitas Tanaman Menggunakan Algoritma Convolutional Neural Network (Cnn).* J. Inform. dan Tek. Elektro Terap, 2023. **11**(3): p. 2830-7062.

12. Susanto, Y.R.W.N., *Analisis sentimen persepsi pengguna layanan aplikasi peduli lindungi menggunakan algoritma support vector machine (studi kasus: aplikasi peduli lindungi pada situs google play).* 2022.

13. Husada, H.C. and A.S. Paramita, *Analisis Sentimen Pada Maskapai Penerbangan di Platform Twitter Menggunakan Algoritma Support Vector Machine (SVM).* Teknika, 2021. **10**(1): p. 18-26.

14. Paramitha, A.A., I. Indriati, and Y.A. Sari, *Analisis Sentimen Terhadap Ulasan Pengguna MRT Jakarta Menggunakan Information Gain dan Modified K-Nearest Neighbor.* Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer, 2020. **4**(4): p. 1125-1132.

15. Ein, I., I. Ernawati, and Y. Widiastiwi. *Analisis sentimen terhadap layanan transjakarta pada media sosial instagram menggunakan naïve bayes dan seleksi fitur information gain*. in *Prosiding Seminar Nasional Mahasiswa Bidang Ilmu Komputer dan Aplikasinya*. 2022.

16. Larasati, K.A., *Analisis Sentimen Masyarakat Pada Media Sosial Twitter Terhadap Sistem Kerja Work From Anywhere (Wfa) Menggunakan Naïve Bayes Dan Particle Swarm Optimization (Pso).*

17. Irnawati, O. and K. Solecha, *Analisis Sentimen Ulasan Aplikasi Flip Menggunakan Naïve Bayes dengan Seleksi Fitur PSO.* Jurnal Ilmiah Intech : Information Technology Journal of UMUS, 2022. **4**(02): p. 189-199.