Scientific Journal of Informatics
Vol. 12, No. 1, Feb 2025

p-ISSN 2407-7658 https://journal.unnes.ac.id/journals/sji/index e-ISSN 2460-0040

Sentiment Analysis of Public Opinion on BAWASLU Using Random Forest and Particle Swarm Optimization

Meida Cahyo Untoro[1*], Muhammad Farhan[2]
[1, 2]Department of Informatics Engineering, Institut Teknologi Sumatera, Indonesia

Abstract.
Purpose: Sentiment analysis, commonly referred to as opinion mining, involves the study of people's opinions, emotions, and attitudes toward various subjects. While the Random Forest algorithm is frequently employed in sentiment classification tasks, its integration with Particle Swarm Optimization (PSO) for feature selection remains relatively underexplored. This study investigates whether PSO-based feature selection can enhance the predictive performance of Random Forest by optimizing the selection of relevant textual features, ultimately leading to more accurate sentiment classification.
Methods: The research adopts a structured text preprocessing approach that includes data cleansing, case folding, normalization, stop-word removal, and stemming to refine the input text. Term Frequency-Inverse Document Frequency (TF-IDF) is applied to extract features, followed by PSO-driven feature selection to refine the input set for the Random Forest classifier. The proposed model is evaluated using a Twitter sentiment dataset related to "Bawaslu", with performance measured based on Out-of-Bag (OOB) error and accuracy metrics.
Result: Empirical results demonstrate that incorporating PSO-based feature selection into the Random Forest model substantially lowers the OOB error to 20.42%, compared to 28.72% in the baseline Random Forest model. Furthermore, the optimized model achieves an accuracy of 78.35%, outperforming the standard approach. However, the introduction of PSO-based feature selection increases computational demands, indicating a trade-off between classification accuracy and processing efficiency.
Novelty: This study introduces the novel integration of PSO-driven feature selection with Random Forest classification for sentiment analysis, addressing challenges in imbalanced text data. By optimizing feature selection through a metaheuristic approach, it enhances model accuracy and efficiency. The novelty lies in applying PSO to refine feature selection in text classification, offering new insights into improving machine learning models for imbalanced datasets. Future research could explore reducing computational overhead and investigating hybrid selection techniques to further enhance scalability and performance.

Keywords: Sentiment analysis, Random forest, PSO, Feature selection
Received March 2025 / **Revised** May 2025 / **Accepted** May 2025

INTRODUCTION

Action Supervisory Board (Bawaslu) is an independent body mandated by Law Number 7 of 2017 on Elections to oversee the electoral process in Indonesia [1]. Its establishment was driven by widespread concerns regarding election violations and disputes, necessitating the formation of an impartial oversight body [2]. In 2020, Bawaslu played a critical role in supervising simultaneous regional elections across various provinces, ensuring that electoral laws were enforced [3]. However, this role has elicited diverse public reactions, ranging from support and appreciation to criticism and skepticism regarding its actions [4]. With the expanding influence of social media, particularly platforms like Twitter, analyzing public sentiment toward Bawaslu's performance provides valuable insights into public trust and perception [5]. Given the sheer volume of election-related discussions on Twitter, manually evaluating sentiment would be inefficient and prone to bias, reinforcing the need for automated sentiment analysis techniques [6].

As one of the most widely used social media platforms in Indonesia, Twitter boasts approximately 556 million members as of early 2023 active users, according to Statistic [7]. The platform allows individuals to share opinions in the form of tweets, often incorporating hashtags to categorize discussions and track trending topics [8]. This creates a vast repository of data [9], offering opportunities for sentiment analysis to assess public opinion toward Bawaslu [10]. Sentiment analysis, also known as opinion mining, is a

[*]Corresponding author.
Email addresses: cahyo.untoro@if.itera.ac.id (Untoro)
DOI: 10.15294/sji.v12i1.22234

branch of artificial intelligence dedicated to analyzing people’s emotions, evaluations, and perspectives regarding specific entities such as organizations, individuals, or events [11]. Sentiments are generally classified into binary categories (positive or negative), ternary categories (including neutral), or more granular ordinal classifications [12]. Given the exponential growth of election-related discourse on Twitter, traditional manual approaches to sentiment classification are no longer practical, necessitating the adoption of machine learning-based sentiment analysis methods [13].

In sentiment analysis, classification algorithms are used to determine the polarity of user opinions. These algorithms typically follow one of two approaches: supervised learning, which relies on pre-labeled training data, and unsupervised learning, which identifies patterns without predefined labels [14]. Recent advancements highlight the effectiveness of feature selection techniques, such as Particle Swarm Optimization (PSO), in enhancing classification performance [15]. PSO has been successfully applied to various machine learning models—including Support Vector Machines (SVM), Decision Trees, and Naïve Bayes—to improve sentiment classification accuracy [16]. Existing studies in 2022 have explored PSO-optimized classification techniques, such as SVM with Principal Component Analysis (PCA), Decision Tree-based sentiment classification, and PSO-enhanced SVM and K-Nearest Neighbors (KNN) models [17].

Random Forest is a well-established machine learning algorithm comprising multiple decision trees, where each tree contributes to the final classification outcome through a majority voting mechanism [18]. The algorithm is widely recognized for its robustness, noise resistance, and ability to handle large-scale datasets efficiently [19]. Compared to boosting techniques like Adaptive Boosting (AdaBoost), RF offers superior accuracy while maintaining faster computational times than both Bagging and Boosting approaches [20]. Given the proven effectiveness of PSO in optimizing feature selection and the high classification accuracy of Random Forest, this study seeks to integrate PSO-based feature selection with RF for sentiment analysis [21]. By combining these techniques, the study aims to enhance the accuracy of Twitter-based sentiment classification related to public opinions on Bawaslu, offering a scalable and efficient approach to large-scale sentiment analysis.

METHODS

The research methodology delineates the systematic stages undertaken in developing a Feature Selection Particle Swarm Optimization (PSO)-Based Sentiment Analysis System using the Random Forest classification approach. This structured framework encompasses sequential phases, including data collection, text preprocessing, dataset partitioning, classification, and model evaluation. Each stage plays a pivotal role in ensuring that the sentiment analysis model efficiently processes and accurately classifies tweets related to Bawaslu. The subsequent sections provide an in-depth explanation of each phase (Figure

Data collection

The initial phase in the sentiment analysis system workflow is data collection. From November 3 to December 9, 2020, data obtained through the Twitter API amounted to 30.8 thousand. The data consists of the date of the post, username, language, and text. The cleaning of the 30.8 thousand data is done against out-of-context, news, duplicate, and retweet data. This study retrieves user-generated content from Twitter by leveraging the Twitter API, which grants access to publicly available tweets. The dataset is curated based on tweets containing the keywords "#BawasluTegasMinimPelanggar" and "Bawaslu", ensuring that the collected data remains relevant to discussions surrounding Bawaslu.

Once the raw data is acquired, the next crucial step involves manual annotation, where each tweet is categorized based on its sentiment polarity. The classification follows a binary sentiment labeling scheme, as outlined below:

Table 1. Confusion matrix

Predicted \ Actual	Positive	Negative
Positive	True Positive	False Positive
Negative	False Negative	True Negative

This labeling process is fundamental to structuring the dataset, which subsequently serves as input for the sentiment analysis system in the following stages of processing.

Text preprocessing

Following data collection, a text preprocessing phase is conducted to clean and standardize the textual data. This step is essential because raw Twitter data often contains noisy elements, such as special characters, URLs, emojis, and inconsistent capitalization, which can compromise the accuracy of sentiment classification. To address these issues, the preprocessing workflow consists of the following steps:

1. Cleansing – Eliminates unnecessary components such as emoticons, Unicode characters, non-alphabetical symbols, usernames, hashtags, and URLs, ensuring that only meaningful text remains.
2. Case Folding – Converts all text into lowercase to maintain uniformity and avoid duplicate word representations.
3. Normalization – Standardizes informal words and abbreviations, while correcting spelling errors and colloquial expressions that could affect data consistency.
4. Stop-word Removal – Filters out frequently occurring words that carry minimal sentiment information (e.g., “in”, “and”, “me”, “this”, “that”). Additionally, words with fewer than four letters are excluded to enhance data relevance.
5. Stemming – Reduces words to their root forms to unify variations of the same word (e.g., “running” → “run”), ensuring better feature representation.

These preprocessing techniques enhance data quality, reduce redundancy, and refine input features, enabling more effective sentiment classification.

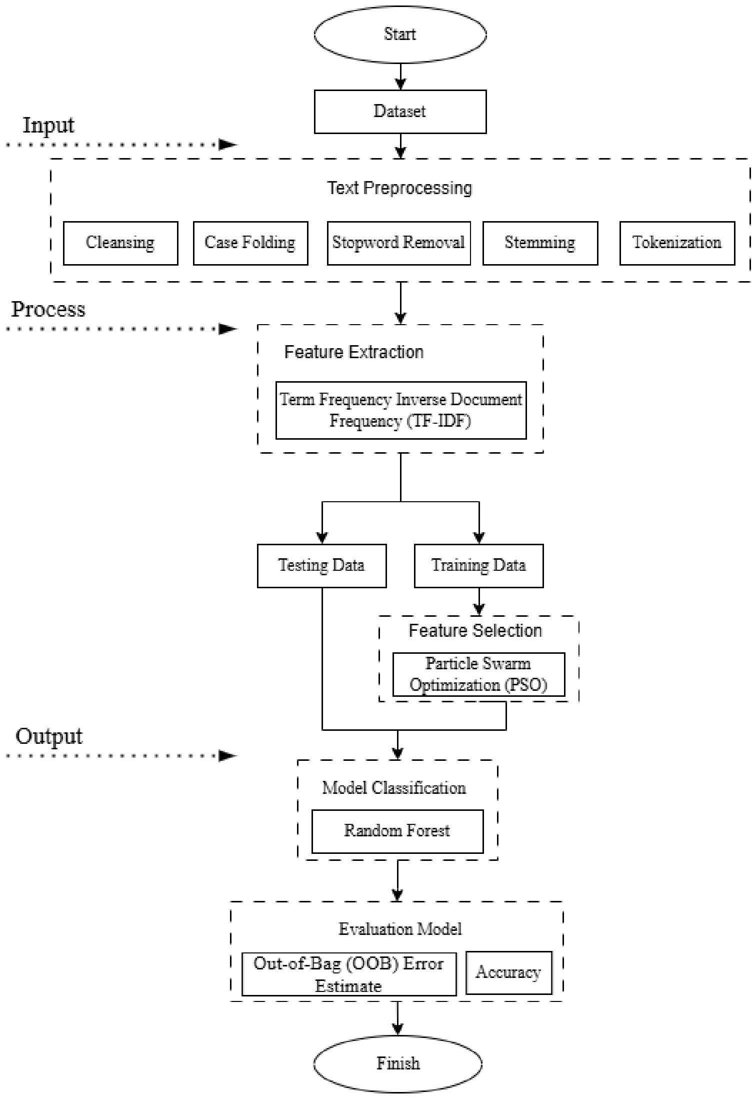

Figure 2. Research methodology flowchart

Splitting the dataset

After completing the preprocessing stage, the dataset is partitioned into training and testing sets to facilitate model learning and performance evaluation. This division enables the classification model to extract patterns from labeled data during training and subsequently assess its predictive capability on unseen data. The dataset is divided as follows [22]:

1. 75% for training – Utilized to train the Random Forest model while optimizing feature selection through Particle Swarm Optimization (PSO).
2. 25% for testing – Allocated for evaluating the trained model's ability to accurately classify previously unseen tweets.

This split ratio ensures that the model receives enough data for training, while maintaining an adequate proportion for reliable performance assessment. The study concludes that the splitting strategy should depend on the dataset size and recommends that dataset splitting is critical for accurate model evaluation and optimal performance.

Classification models

This study constructs two classification models for sentiment analysis to evaluate the impact of PSO-based feature selection on classification performance:

1. Random Forest without PSO-based Feature Selection
    a. In this model, Random Forest is applied directly to the dataset without prior optimization of feature selection.
2. Random Forest with PSO-based Feature Selection
    a. In this approach, Particle Swarm Optimization (PSO) is integrated with Random Forest, enabling the identification of optimal features before classification.
    b. The PSO algorithm iteratively analyzes the dataset to determine the most relevant features, enhancing the classifier's ability to differentiate between positive and negative sentiments.

By comparing these two models, this study aims to determine whether PSO-based feature selection improves the performance of sentiment classification using the Random Forest method.

Evaluation metrics

To assess the effectiveness of the Random Forest classification models, this study employs two key evaluation metrics:

1. The Out-of-Bag (OOB) Error Estimate is chosen as an evaluation metric for the Random Forest classification model for several reasons [23]:
    - Internal Validation: The OOB error estimate provides an internal validation method within the Random Forest algorithm itself, allowing the model to assess its performance without the need for a separate validation set. This is particularly useful when limited data is available, as it maximizes the use of all training samples.
    - Model Reliability: A lower OOB error value indicates better classification performance, meaning the model has effectively learned to generalize from the data and is making fewer errors in its predictions. Conversely, a higher OOB error suggests that the model is not performing as well and may suffer from issues such as overfitting or underfitting, reducing its reliability.
    - Efficiency: By using the OOB error estimate, Random Forests inherently perform cross-validation on each tree built during the training process. This reduces the computational overhead of manually splitting the data into training and validation sets, offering a more efficient way to evaluate the model's performance.
2. Accuracy
    Measured using a Confusion Matrix, which compares actual sentiment labels with model-predicted labels [24].

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

By comparing OOB error and accuracy values between the two models, this study evaluates whether PSO-based feature selection contributes to improved classification performance.

Evaluation

This study introduces a systematic methodology for sentiment analysis, leveraging Particle Swarm Optimization (PSO) for feature selection in conjunction with the Random Forest classification algorithm. The workflow encompasses data collection via the Twitter API, text preprocessing, dataset partitioning, classification, and performance evaluation. By integrating PSO with Random Forest, the study seeks to optimize feature selection, thereby enhancing classification accuracy.

Through this approach, the research offers valuable insights into the effectiveness of feature selection techniques in sentiment analysis. Additionally, it examines whether the PSO-optimized Random Forest model surpasses the performance of the standard Random Forest classification. The findings contribute to the ongoing exploration of machine learning-driven sentiment analysis, highlighting the potential advantages of metaheuristic-based feature selection in text classification tasks.

RESULTS AND DISCUSSIONS

Data obtained through the Twitter API from November 3, 2020 to December 9, 2020 amounted to 30.8 thousand data. The dataset includes several attributes: the posting date, username, language, and text content of the posts. To ensure the relevance of the data for sentiment analysis, a data cleaning process was carried out. This process involved removing irrelevant data, such as news articles, duplicate entries, and retweets. For sentiment analysis, only the text content of the posts was retained, while other attributes, including the posting date, username, and language, were discarded. Each remaining data entry was subsequently assigned a label or class for further analysis and distribution data 52.85% class 1 and class 0 47.15% (Table 1).

Table 2. Dataset

No	Text	Class
1	@kabarklaten Jangan bnyk berharap pd @bawaslu_klaten BELGEDES, Bawaslu kie kudu di demo sak kabupaten kie Ben tegas kerjane	0
2	@kafiradikalis @Chilli_Pari Bawaslu nya priksa dan gol kan skalian	0
…	…	…
…	…	…
385	@PDI_Perjuangan @Acuantodaycom Laporkan setiap pelanggaran kampanye k Bawaslu agar Pilkada sehat	1
386	@Aryprasetyo85 @kemendagri @jokowi @KPU_ID @bawaslu_RI Mantul ketegasannya #PaslonAbaiProkesDipidana	1

Once the dataset is obtained and inputted into the machine, the dataset will then go through the text preprocessing process. In the text preprocessing process starts from the cleansing process and case folding (Table 2). In the cleaning and case folding section it is an easy step to do by using regular expression, string and html libraries.

Table 3. Cleaning and case folding

No	Text	Cleaning and Case folding
1	@kafiradikalis @Chilli_Pari Bawaslu nya priksa dan gol kan skalian	bawaslu nya priksa dan gol kan skalian
2	@kabarklaten Jangan bnyk berharap pd @bawaslu_klaten BELGEDES, Bawaslu kie kudu di demo sak kabupaten kie Ben tegas kerjane	jangan bnyk berharap pd belgedes bawaslu kie kudu di demo sak kabupaten kie ben tegas kerjane
…	…	…
…	…	…
385	@Aryprasetyo85 @kemendagri @jokowi @KPU_ID @bawaslu_RI Mantul ketegasannya #PaslonAbaiProkesDipidana	mantul ketegasannya
386	@CNNIndonesia Laporin aja ke BAWASLU biar ditindak	laporin aja ke bawaslu biar ditindak

In the text preprocessing section of normalization, informal words are changed to formal form done with the help of Colloquial Indonesia Lexicon which has resulted from previous research. Each word in the document will be checked if the word includes informal words contained in the dictionary and will be converted into a formal word if found. The results of the normalization process (Table 3). In the stop word section, it is done using NLTK and Sastrawi libraries coupled with additional words obtained from observations and words that are less than 4 characters [25]. While the stemming part only uses Sastrawi library because NLTK does not support the Indonesian stemming process.

Table 4. Normalization

No	Cleaning and Case folding	Normalization
1	jangan bnyk berharap pd belgedes bawaslu kie kudu di demo sak kabupaten kie ben tegas kerjane	jangan banyak berharap percaya diri belgedes bawaslu kie kudu di demo sak kabupaten kie ben tegas kerjane
2	bawaslu nya priksa dan gol kan skalian	bawaslu nya priksa dan gol kan skalian
…	…	…
…	…	…
385	laporkan segala bentuk pelanggaran pilkada kepada bawaslu	laporkan segala bentuk pelanggaran pilkada kepada bawaslu
386	laporin aja ke bawaslu biar ditindak	laporin saja ke bawaslu biar ditindak

The stop words removal and stemming process is a form of normalization. Each word in the document is checked against a stop word dictionary; if found, it is removed. If a word matches a stemming dictionary, it is reduced to its base form. The results of the stop word and stemming process (Table 4). In this study the tokenization process was conducted at once during the feature extraction process because the library used supports the ngram tokenization process. At the end of this process, visualization of 20 words with the most frequency that appears on the dataset Figure 2.

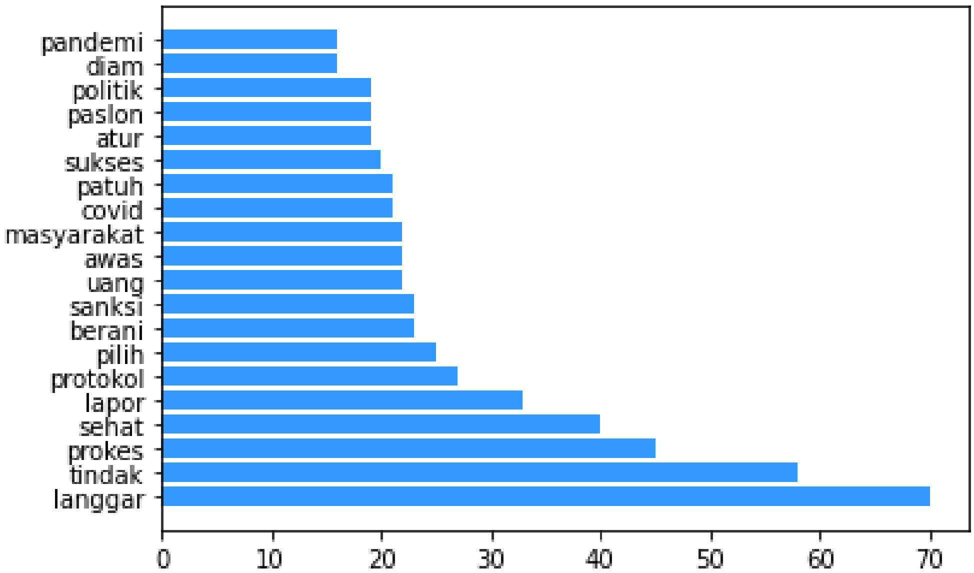

Figure 1. Graph of word occurrences

Feature extraction

The study leverages TF-IDF (Term Frequency-Inverse Document Frequency) for feature extraction, using the TfidfVectorizer function from scikit-learn. This function processes textual data by transforming it into numerical representations that capture the importance of each word or token in the corpus. The process is parameterized by several settings: ngram range, minimum document frequency (DFT), and maximum DFT. In this study, n-gram tokens are extracted from unigrams (1-gram) to trigrams (3-gram), which means that the tokenization includes individual words as well as common word pairs (bigrams) and triplets (trigrams). This enables the model to capture contextual relationships between consecutive words, which is crucial for many natural language processing tasks like sentiment analysis and topic modeling. For example, bigrams like "abai prokes" and "safe comfortable," and trigrams such as "safe comfortable intervene," help in capturing more complex patterns that unigrams alone might miss.

Additionally, the DFT settings are configured with a minimum occurrence threshold of 2 and a maximum DFT set to half the total number of documents in the corpus. This ensures that the extracted features are not overly sparse or too common, both of which might not contribute significantly to distinguishing between classes. After feature extraction, a total of 457 columns (features) and 386 rows (samples) are generated, forming the feature matrix used for further analysis and modeling.

For feature selection, the study employs Particle Swarm Optimization (PSO), specifically using the BPSO (Binary PSO) variation from the pswarms library. PSO is an optimization algorithm inspired by the social behavior of birds and fish, where each particle (solution) adjusts its position based on both its own experience and the experiences of its neighbors. Each particle is evaluated based on its ability to minimize the Out-of-Bag (OOB) error for the feature subset it represents. In the binary representation of each particle, a "1" indicates that a feature is selected, while "0" indicates that the feature is not selected. The optimization task thus becomes finding the best configuration of features (represented as binary vectors) that minimizes the error.

Table 5. Stop ward and steaming

No	Normalization	Stop ward and Steaming
1	jangan banyak berharap percaya diri belgedes bawaslu kie kudu di demo sak kabupaten kie ben tegas kerjane	harap percaya belgedes kudu demo kabupaten kerjane
2	dan tak satupun laporan pelanggaran ditindaklanjuti dengan hukuman bawaslu cuma melengkapi struktur formalitas	satu lapor langgar ditindaklanjuti hukum lengkap struktur formalitas
…	…	…
…	…	…
385	laporkan segala bentuk pelanggaran pilkada kepada bawaslu	lapor bentuk langgar
386	mantul ketegasannya	mantul tegas

```
best pos:
 [1 0 0 1 0 0 0 1 1 0 1 1 1 1 0 1 0 0 0 1 0 0 1 0 1 1 0 0 0 0 1 0 1 1 0 0 0
 0 0 1 1 0 1 1 0 0 1 1 1 0 0 1 0 1 0 1 0 1 1 0 1 1 1 0 1 1 1 0 0 1 1 0 0 1
 0 1 1 1 0 1 0 0 0 0 0 1 0 1 1 1 0 1 0 0 0 0 0 1 0 0 1 1 0 0 0 0 1 1 0 0 1
 0 1 1 0 0 0 1 1 1 0 0 0 1 0 0 0 0 0 1 0 0 1 0 1 0 0 0 0 0 1 1 1 0 1 0 1 1
 0 1 1 1 1 0 1 0 1 1 1 0 1 0 1 1 0 0 1 1 1 1 1 1 0 1 1 0 1 1 1 0 0 0 0 1 1
 0 1 1 1 1 1 1 0 0 0 1 1 1 1 1 0 1 0 0 1 0 1 0 0 1 1 1 1 1 1 1 0 1 0 1 1 0
 0 0 0 1 1 0 0 1 0 1 1 1 1 1 0 1 1 0 0 1 1 1 0 1 0 1 1 0 1 1 1 0 1 0 1 1 1
 1 0 1 1 1 1 1 1 0 1 1 1 1 1 0 1 0 1 1 0 0 0 1 1 1 0 1 0 0 1 0 1 0 1 1 0 1
 0 0 1 1 0 1 1 0 0 0 0 1 1 1 1 0 0 0 0 0 1 0 0 1 1 1 1 0 1 0 0 0 0 0 0 1 1
 0 0 0 0 1 0 0 1 0 0 0 1 0 1 0 1 1 1 1 1 0 1 0 0 1 0 1 0 1 0 0 0 0 0 1 1 0
 0 0 1 1 0 1 1 1 0 1 1 1 0 1 0 1 0 1 0 1 0 1 1 1 0 1 0 0 1 0 1 0 0 1 0 1 1
 1 1 1 0 0 1 1 1 0 1 0 1 0 1 0 0 0 0 0 0 0 1 1 1 0 0 0 0 1 0 0 0 0 1 1 0 1
 0 0 0 1 0 0 0 0 0 0 0 1 0]
```

Figure 2. Best particle position

The PSO algorithm progresses through iterations, with the fitness value indicating how well the particle performs. The graph of fitness values shows a stepwise reduction in error, suggesting that the algorithm is successfully refining its solution with each iteration. As particles converge towards the optimal feature subset, the final position (solution) provides the best configuration for feature selection.

Table 6. Feature

No	Feature	No	Feature
1	abai	…	…
2	abai prokes	…	…
3	acara bubar	…	…
4	wajib tindak	455	antisipasi
5	tunggu lapor	456	antusias
…	…	457	antusiasme

The process of feature selection Particle Swarm Optimization is done using pswarms library that provides BPSO variation. The BPSO function requires input parameters in the form of swarm size, search dimensions and other settings such as inertia weight, acceleration coefficient and maximum particle neighbor.

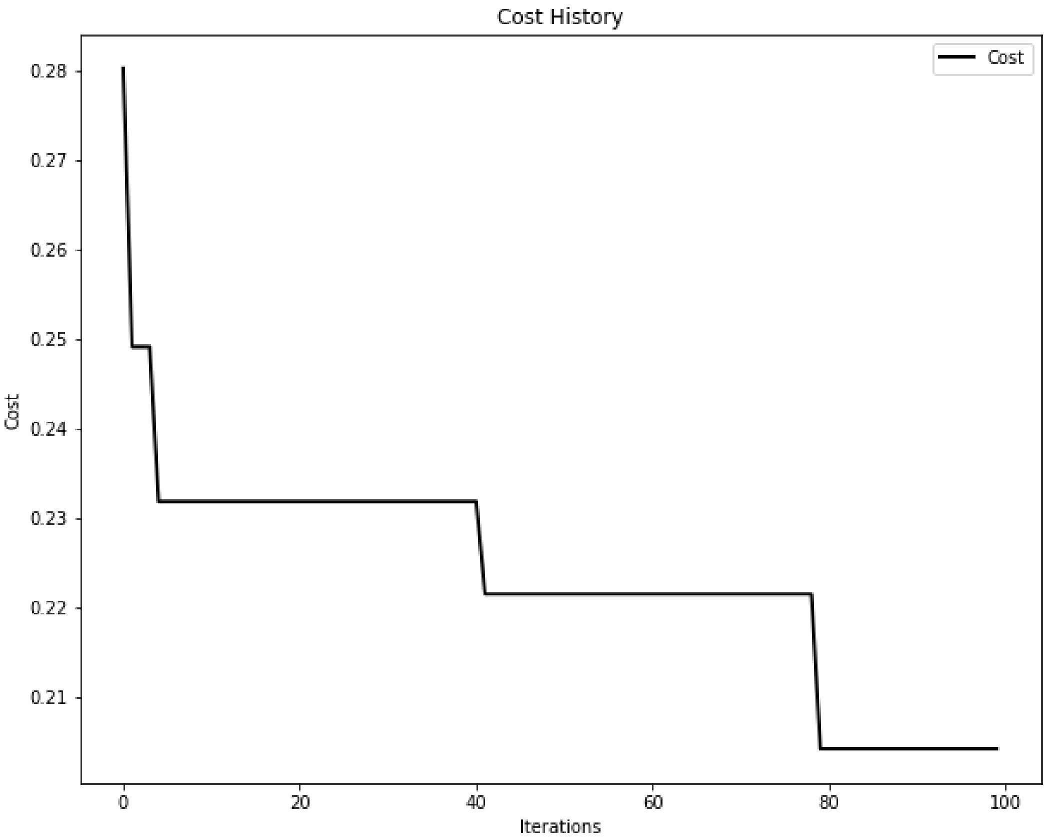

Figure 3 Cost fitness value on PSO

Evaluation

Table 7. Feature extraction using TF.IDF

	0	1	2	3	…	…	455	456	457
0	0	0	0,9	0	…	…	0,28	0	0
1	0	0	0	0,6	…	…	0	0,62	0
2	0	0	0	0	…	…	0	0	0
3	0	0	0	0	…	…	0	0	0
…	…	…	…	…	…	…	…	…	…
…	…	…	…	…	…	…	…	…	…
455	0	0	0,29	0	…	…	0	0	0
456	0	0	0	0,61	…	…	0	0,35	0
457	0	0	0	0	…	…	0	0	0,51

Each classification model uses Random Forest with a total of 100 trees. Random Forest classification model without PSO is conducted once experiment while random forest classification model based on PSO is done with iterations of 100 iterations where each experiment is conducted 3 times with parameter settings Table 7. The selection of parameters used is based on the recommended value of PSO usage but there is a test parameter that is of the same value that is the maximum number of particle neighbors as much as 3 particles [26]. This study does not explicitly discuss overfitting, despite its potential impact on model performance. The focus is on feature selection using Particle Swarm Optimization (PSO) and TF-IDF feature extraction. While these techniques aim to optimize and simplify the model, the issue of overfitting is not addressed. To ensure generalization, further steps like cross-validation and regularization could be explored in future research. Evaluating the model's performance on both training and test data would also help identify and mitigate overfitting, ensuring the model is robust and reliable.

Table 8 Result PSO

Parameter	Inertia ($\boldsymbol{\omega}$)	Coefficient ($\boldsymbol{\varphi_1}$)	Time (Sec)	OOB Error (%)
Iteration 1	0,7	1,43	6,2	22,84
Iteration 2	0,7	1,43	9,3	21,80
Iteration 3	0,8	1,62	6,1	22,15

Table 8 the use of Random Forest models based on PSO feature selection works better than Random Forest models without PSOs. PSO-based Random Forest model with 100 iterations obtaining the smallest OOB error value in the 3rd experiment that required an optimization time of 9.0 minutes resulting in an OOB error of 20.42%. A particle and iterations used in PSO can increase fitness value results in direct proportion to lama optimization time required.

Table 9. Evaluation of random forest

Feature Selection	Time (Sec)	OOB Error (%)
Without PSO	12	28,72
PSO	9,0	20,42

Table 10. Result dataset testing

No	Text	Class	Without PSO	PSO
1	lengkap laksana daya hadap calon usung kuasa	0	0	1
2	berani periksa langgar	0	0	1
…	…	…	…	…
…	…	…	…	…
96	sepakat lemah sawar lemah langgar indonesia	1	1	0
97	langgar protokol sehat sanksi paslon kepala	1	0	1

After the model is completed, predictions are made to see the classification of the model in handling testing data or outside the training data. Table 9 test data predictions results where the column "class" is the actual class (actual class) while the other right column is the prediction result of the previous test model. Random Forest model prediction without PSO based on confusion matrix Table 10, the correct data predicted (TP and TN) of 68 out of 97 data resulted in an accuracy value of 70.10%. Meanwhile, the PSO-based Random Forest model was tested using a subset feature that has the lowest OOB value 100 iterations. The correct data predicted by 75 out of 97 data resulted in an accuracy value of 77.32%. As well as predictions of PSO-based Random Forest models with 100 iterations based on confusion matrix (Table 12), the correct data predicted by 76 out of 97 data produces an accuracy value of 78.35%. The accuracy of the Random Forest model, both with and without the Particle Swarm Optimization (PSO) technique, remains below 0.8, despite some improvements with PSO. Several factors could explain this outcome. Firstly, the inherent complexity of the data may exceed the model's ability to capture relevant patterns. While Random Forest is an ensemble learning algorithm that performs well with high-dimensional data, it may struggle with intricate, non-linear relationships that require more advanced models like deep learning. Additionally, although PSO aids in feature selection, the selected features may not fully capture the necessary information for optimal performance.

This suggests that the feature subset chosen by PSO might still be suboptimal. Another potential factor is the quality of the data itself. If the data contains noise, missing values, or class imbalances, these issues can impede the model's ability to learn effectively, limiting its predictive accuracy. Furthermore, the model may be underfitting, particularly if the selected features fail to capture the full complexity of the data. Alternatively, overfitting could also occur if too many irrelevant or overly specific features are chosen, leading to a model that does not generalize well to unseen data. Hyperparameter tuning may also be necessary, as Random Forest models often benefit from adjustments such as the number of trees, the maximum depth, and the minimum samples per leaf. If these hyperparameters are not properly optimized, the model may not reach its full potential. Additionally, while PSO was run for 100 iterations, this may not have been sufficient to converge on an optimal feature set. Increasing the number of iterations or fine-tuning the PSO parameters could further improve results. Lastly, data preprocessing steps, such as cleaning, imputation, and augmentation, could enhance the model's ability to identify patterns. To improve accuracy, further iteration of the PSO algorithm, along with more complex models, hyperparameter tuning, and comprehensive data preprocessing, may be necessary to achieve better results and surpass the 0.8 accuracy threshold. PSO's effectiveness in enhancing model performance by reducing noise and selecting relevant features.

Table 11. Confusion matrix without PSO

		Predicted Class		Total
Actual Class	Positive	41	12	53
	Negative	17	27	44
Total		58	39	97

Table 12. Confusion matrix PSO

		Predicted Class		Total
Actual Class	Positive	40	13	53
	Negative	8	36	44
Total		48	49	97

The results of the predictive accuracy value of each test model can be seen in the summary Table 13. The accuracy of the Random Forest model was improved by Particle Swarm Optimization (PSO) for feature selection. The model achieved 70.10% accuracy without PSO and 78.35% with PSO, demonstrating that PSO enhanced model performance. Similar improvements have been reported in existing research. Showed a 5-10% accuracy increase using PSO for feature selection and found PSO improved classification accuracy by optimizing feature sets. These results, aligning with previous studies, highlight.

Table 13. Accuracy random forest

Feature Selection	Accuracy (%)
Without PSO	70,10
PSO	78,35

CONCLUSION

The sentiment analysis research using Particle Swarm Optimization (PSO) feature selection combined with the Random Forest method leads to several key conclusions based on the experiments conducted. The PSO-based feature selection significantly improves the Out-of-Bag (OOB) error rate, reducing it to 20.42%, compared to 28.72% in the standard Random Forest model without PSO. This demonstrates that incorporating PSO enhances model performance by selecting the most relevant features. Additionally, the PSO-based Random Forest model effectively classifies sentiment in Twitter users' opinions about Bawaslu, achieving a best accuracy score of 78.35%. While the PSO-enhanced model yields superior classification results, it has a notable drawback—longer processing times due to the computational complexity of the feature selection process. Increasing the swarm size and the number of iterations can further improve fitness values, but this improvement comes at the cost of significantly higher computational time for the optimization process.

REFERENCES

[1] D. Setiawan, H. Sejati, and N. A. Zaini, "Strengthening the Supervision Authority of Election Socialization through Collaboration Between KPU , Bawaslu , and Regional Governments," *Cogn. Civitatis Polit.*, vol. 1, no. 5, 2024.

[2] Pujiono and N. Prasetyoningsih, "ELECTIONS FOR SALE : THE SHORTCOMINGS OF INDONESIA ' S ELECTION LAW IN PREVENTING VOTE-BUYING AND ENSURING FAIR PLAY," vol. 3, no. 2, pp. 1043–1066, 2024.

[3] N. N. Mutiarasari and R. Herawati, "Supervision of Bawaslu Pemalang Regency in the 2020 Regional Head Election," *LAW REFORM*, vol. 16, no. 2, pp. 264–275, Sep. 2020, doi: 10.14710/lr.v16i2.33777.

[4] M. Wahyudi, B. Ngarawula, and B. Prianto, "Generation Z Voter Behavior (Case Study of Generation Z Voter Behavior Ahead of the 2024 Surabaya City Regional Head Election)," *Int. J. Res. Soc. Sci. Humanit.*, vol. 05, no. 12, pp. 10–35, 2024, doi: 10.47505/IJRSS.2024.12.2.

[5] K. Hasan, Z. Zulfadli, M. Muchlis, M. Masriadi, A. Husna, and A. Awaluddin, “Political Public Space In The 2024 Election Social Media Platform; Between Expectations And Reality,” *Proc. Int. Conf. Soc. Sci. Polit. Sci. Humanit.*, vol. 4, p. 00031, Jan. 2024, doi: 10.29103/icospolhum.v4i.408.
[6] A. Deb, L. Luceri, A. Badaway, and E. Ferrara, “Perils and Challenges of Social Media and Election Manipulation Analysis: The 2018 US Midterms,” in *Companion Proceedings of The 2019 World Wide Web Conference*, New York, NY, USA: ACM, May 2019, pp. 237–247. doi: 10.1145/3308560.3316486.
[7] V. F. Hakim and D. Riana, “Analysis of User Complaints for Telecommunication Brands on X (Twitter) using IndoBERT and Deep Learning,” *J. Nas. Pendidik. Tek. Inform.*, vol. 13, no. 2, pp. 270–279, Jul. 2024, doi: 10.23887/janapati.v13i2.76497.
[8] B. (Kevin) Chae, “Insights from hashtag #supplychain and Twitter Analytics: Considering Twitter and Twitter data for supply chain practice and research,” *Int. J. Prod. Econ.*, vol. 165, pp. 247–259, Jul. 2015, doi: 10.1016/j.ijpe.2014.12.037.
[9] S. Juanita, “Analisis Sentimen Persepsi Masyarakat Terhadap Pemilu 2019 Pada Media Sosial Twitter Menggunakan Naive Bayes,” *J. MEDIA Inform. BUDIDARMA*, vol. 4, no. 3, p. 552, Jul. 2020, doi: 10.30865/mib.v4i3.2140.
[10] R. Zulfauzan and A. W. S. Untung, “Political Marketing Strategies of the Golkar Party’s Regional Leadership Council in West Kotawaringin Regency for the 2024 Election,” *Society*, vol. 13, no. 1, pp. 144–168, Mar. 2025, doi: 10.33019/society.v13i1.794.
[11] M. Wankhade, A. C. S. Rao, and C. Kulkarni, “A survey on sentiment analysis methods, applications, and challenges,” *Artif. Intell. Rev.*, vol. 55, no. 7, pp. 5731–5780, Oct. 2022, doi: 10.1007/s10462-022-10144-1.
[12] B. Andrian, T. Simanungkalit, I. Budi, and A. F. Wicaksono, “Sentiment Analysis on Customer Satisfaction of Digital Banking in Indonesia,” *Int. J. Adv. Comput. Sci. Appl.*, vol. 13, no. 3, 2022, doi: 10.14569/IJACSA.2022.0130356.
[13] F. Yi, H. Liu, H. He, and L. Su, “A Comparative Analysis of Active Learning for Rumor Detection on Social Media Platforms,” *Appl. Sci.*, vol. 13, no. 22, p. 12098, Nov. 2023, doi: 10.3390/app132212098.
[14] M. C. Untoro *et al.*, “Evaluation of cosine similarity and deci similarity on students’ test essay answers dataset,” 2024, p. 020013. doi: 10.1063/5.0205162.
[15] T. M. Shami, A. A. El-Saleh, M. Alswaitti, Q. Al-Tashi, M. A. Summakieh, and S. Mirjalili, “Particle Swarm Optimization: A Comprehensive Survey,” *IEEE Access*, vol. 10, pp. 10031–10061, 2022, doi: 10.1109/ACCESS.2022.3142859.
[16] E. Elgeldawi, A. Sayed, A. R. Galal, and A. M. Zaki, “Hyperparameter Tuning for Machine Learning Algorithms Used for Arabic Sentiment Analysis,” *Informatics*, vol. 8, no. 4, p. 79, Nov. 2021, doi: 10.3390/informatics8040079.
[17] S. Agrawal, S. K. Jain, A. Khatri, M. Agarwal, A. Tripathi, and Y.-C. Hu, “Novel PSO Optimized Voting Classifier Approach for Predicting Water Quality,” *Math. Probl. Eng.*, pp. 1–14, Jul. 2022, doi: 10.1155/2022/6445580.
[18] H. Dabiri, V. Farhangi, M. J. Moradi, M. Zadehmohamad, and M. Karakouzian, “Applications of Decision Tree and Random Forest as Tree-Based Machine Learning Techniques for Analyzing the Ultimate Strain of Spliced and Non-Spliced Reinforcement Bars,” *Appl. Sci.*, vol. 12, no. 10, p. 4851, May 2022, doi: 10.3390/app12104851.
[19] A. Chaudhary, S. Kolhe, and R. Kamal, “An improved random forest classifier for multi-class classification,” *Inf. Process. Agric.*, vol. 3, no. 4, pp. 215–222, Dec. 2016, doi: 10.1016/j.inpa.2016.08.002.
[20] D. Adesina, C.-C. Hsieh, Y. E. Sagduyu, and L. Qian, “Adversarial Machine Learning in Wireless Communications Using RF Data: A Review,” *IEEE Commun. Surv. Tutorials*, vol. 25, no. 1, pp. 77–100, 2023, doi: 10.1109/COMST.2022.3205184.
[21] W. Shafqat, S. Malik, K.-T. Lee, and D.-H. Kim, “PSO Based Optimized Ensemble Learning and Feature Selection Approach for Efficient Energy Forecast,” *Electronics*, vol. 10, no. 18, p. 2188, Sep. 2021, doi: 10.3390/electronics10182188.
[22] T. F. Thien and W. S. Yeo, “A comparative study between PCR, PLSR, and LW-PLS on the predictive performance at different data splitting ratios,” *Chem. Eng. Commun.*, vol. 209, no. 11, pp. 1439–1456, Nov. 2022, doi: 10.1080/00986445.2021.1957853.
[23] D. R. Cutler *et al.*, “RANDOM FORESTS FOR CLASSIFICATION IN ECOLOGY,” *Ecology*, vol. 88, no. 11, pp. 2783–2792, Nov. 2007, doi: 10.1890/07-0539.1.
[24] M. C. Untoro and M. A. N. M. Yusuf, “Evaluate of Random Undersampling Method and Majority

Weighted Minority Oversampling Technique in Resolve Imabalanced Dataset," *IT J. Res. Dev.*, vol. 8, no. 1, pp. 1–13, Aug. 2023, doi: 10.25299/itjrd.2023.12412.

[25] Rianto, A. B. Mutiara, E. P. Wibowo, and P. I. Santosa, "Improving the accuracy of text classification using stemming method, a case of non-formal Indonesian conversation," *J. Big Data*, vol. 8, no. 1, p. 26, Dec. 2021, doi: 10.1186/s40537-021-00413-1.

[26] T. H. Pham and B. Raahemi, "Bio-Inspired Feature Selection Algorithms With Their Applications: A Systematic Literature Review," *IEEE Access*, vol. 11, pp. 43733–43758, 2023, doi: 10.1109/ACCESS.2023.3272556.