Journal of Artificial Intelligence and Engineering Applications

Website: https://ioinformatic.org/

15th February 2025. Vol. 4. No. 2; e-ISSN: 2808-4519

The Impact of Principal Component Analysis Dimensionality Reduction on Sentiment Classification Performance Using Support Vector Machine

Azzahra Moudy Fajria[1*], Ahmad Faqih[2], Gifthera Dwilestari[3]

[1, 2]*Informatics Engineering, STMIK IKMI Cirebon*
[3]*Information Systems, STMIK IKMI Cirebon*
[*]*azzahramoudy14@gmail.com*

Abstract

This study investigates the application of Principal Component Analysis (PCA) to enhance sentiment classification performance using the Support Vector Machine (SVM) algorithm. User reviews of the ChatGPT application from the Play Store were collected, preprocessed, and analyzed to identify the sentiment within the text (positive, negative, or neutral). The research follows the Knowledge Discovery in Databases (KDD) framework, starting with data selection, preprocessing, transformation, and applying PCA for dimensionality reduction. PCA was used to reduce the complexity of the high-dimensional text data, improving SVM's efficiency in sentiment classification. Evaluation results show that applying PCA led to an improvement in model performance, with accuracy increasing from 72.65% to 73.20%, precision from 71.58% to 72.24%, recall from 71.77% to 72.66%, and F1-score from 71.56% to 72.32%. Although the improvements were modest, the findings demonstrate that PCA effectively simplifies complex datasets and enhances SVM performance in sentiment classification, offering benefits in processing high-dimensional text data.

Keywords: *Principal Component Analysis, Support Vector Machine, Sentiment Analysis, Dimension Reduction*

1. Introduction

The rapid development of the field of Informatics has had a significant impact on business, education, and public services. Artificial intelligence and data analytics have become crucial for data-driven decision-making. Sentiment analysis facilitates understanding public opinions through unstructured text, such as reviews on social media and e-commerce platforms. Natural language processing (NLP) technology enhances the role of sentiment analysis in automatically predicting opinions and evaluating services. The industrial sector is shifting towards automated systems to respond to consumer sentiments quickly and accurately. The increasing volume of data from social media and consumer reviews demands more efficient methods. User reviews are an efficient and effective tool for obtaining information about products or applications [1]. Therefore, optimizing sentiment analysis techniques through computational approaches is becoming essential in this information era. Sentiment analysis is an automated method for understanding and processing text-based data to extract relevant information. The process aims to identify opinions or viewpoints regarding a particular topic, whether related to individuals, organizations, or products, within a specific dataset [2].

Data mining is the process of extracting or uncovering previously unknown information from large datasets. The resulting information is interpretable, relevant, and valuable for supporting critical business decision-making. This process combines various techniques to identify hidden patterns within collected data. By employing data mining, users can discover knowledge in databases that would not have been detected manually. Furthermore, data mining enables the automated retrieval of significant information from vast data storage systems [3].

Following the discussion of the definition and significance of sentiment analysis in natural language processing, the next step is to consider the challenges faced in sentiment classification, particularly the high dimensionality of the data, which can adversely affect model performance. One effective technique that can be employed is dimensionality reduction through Principal Component Analysis (PCA) to address this issue. Principal Component Analysis (PCA) is a method for transforming vector-based data into a simplified representation. This approach presents the data in a lower-dimensional space while retaining a clear and informative descriptive overview of the original information [4].

The application of Support Vector Machine (SVM) as a classification algorithm is explored in this context. Support Vector Machine (SVM) is a method used to determine the optimal hyperplane, which is a function that separates data into distinct classes. This hyperplane is designed to maximize the margin, defined as the distance between the data points of both classes and the hyperplane, ensuring robust class separation [5]. SVM, known for its ability to separate data with a maximal margin, is an ideal choice for analyzing text data that has undergone dimensionality reduction. By combining PCA and SVM, it is anticipated that optimal sentiment classification performance can be achieved.

Several studies have been conducted on sentiment analysis using the Support Vector Machine (SVM) method and the application of Principal Component Analysis (PCA) across various platforms. One notable study focused on using PCA to address the challenges of segmentation in high-dimensional data, as presented in the journal titled "Pengenalan Pola Batik Lampung Menggunakan Metode Principal Component Analysis". This study analyzed four distinct batik motifs, namely kain sembagi, siger, batik tulis, and gajah & kapal, applied to a dataset consisting of 100 images. PCA was employed to enhance the clarity of pattern details on the batik fabric. The findings of this study demonstrated that the PCA method was highly effective in improving the accuracy of Lampung batik pattern recognition. These results can serve as a foundation for further image classification studies and applications involving batik motif recognition [6].
Furthermore, another study focused on the sentiment classification journal titled "Pengklasifikasian Sentimen Ulasan Aplikasi Whatsapp Pada Google Play Store Menggunakan Support Vector Machine". A total of 1000 reviews were collected through web scraping, with training and testing data split at 90:10 and 80:20 ratios. The study found that using SVM with optimal parameters, such as C=1.0 and γ=1.0, achieved an accuracy of 82%. The findings concluded that the SVM method is effective in classifying user sentiment, particularly in high-dimensional data environments [7].

2. Research Method

This study adopts a quantitative and experimental approach to assess the impact of Principal Component Analysis (PCA) on improving sentiment classification performance using the Support Vector Machine (SVM) algorithm. Knowledge Discovery in Databases (KDD) is discovering hidden information or patterns within a database [8]. Following the Knowledge Discovery in Databases (KDD) framework, the research involves several stages: data selection, preprocessing, transformation, data mining, and model evaluation. Initially, relevant sentiment data is collected and carefully selected, followed by preprocessing steps to eliminate noise and standardize the data format. PCA is then applied to reduce the dimensionality of the data, emphasizing principal components to mitigate overfitting and enhance the efficiency and accuracy of the SVM classifier in distinguishing between positive, neutral, and negative sentiments. The model's performance is evaluated by comparing the evaluation matrix of the SVM with PCA to that without PCA in order to determine the extent to which PCA improves SVM performance.

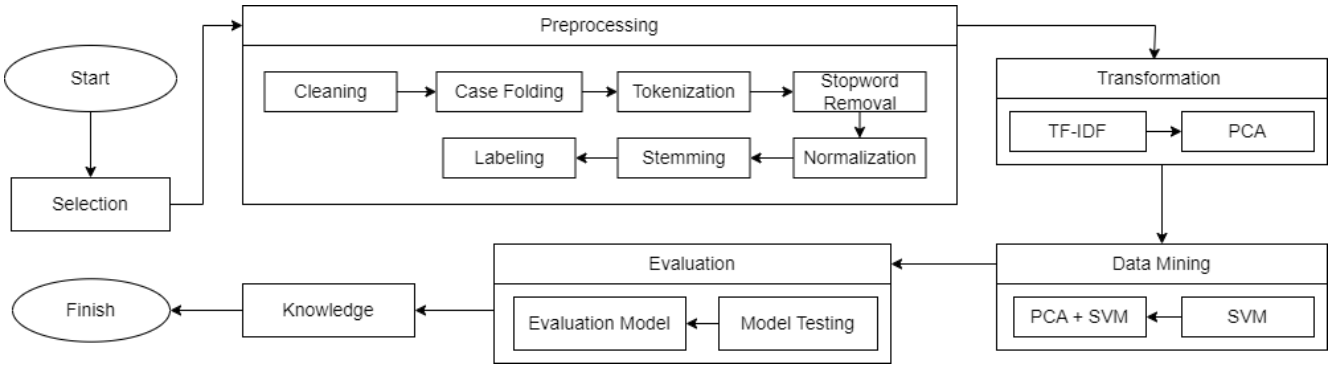

Fig. 1: Research method stages

2.1. Data Selection

The selection phase begins with data collection as the foundation for the analysis process. This study gathered data through web scraping of 10,000 user reviews for the ChatGPT application available on the Play Store. The collected data consists of textual reviews that reflect users' experiences with the application, which will subsequently be processed for sentiment analysis.

2.2. Preprocessing

Preprocessing is a critical stage in data mining, aimed at preparing data for further processing. The data used is often not in an ideal condition, requiring adjustments to address various issues such as missing data, redundancy, anomalies (outliers), or incompatible data formats. This process ensures improved data quality, enabling more accurate and reliable results in data mining [9]. Preprocessing is a crucial stage in the Knowledge Discovery in Databases (KDD) process that focuses on preparing raw data for further analysis. During this phase, various methods are employed to clean and format the data, thereby improving its quality and ensuring that the subsequent analysis yields more accurate results.
In sentiment analysis, the preprocessing process is divided into two main stages: data preprocessing and text preprocessing.

2.2.1. Data Preprocessing

The data preprocessing stage focuses on the overall preparation of the dataset, including checks for consistency and completeness of the data used in the study. Data cleaning removes irrelevant or incorrect elements from the dataset, ensuring that the data is better suited for analysis.

Table 1: Dataset after data pre-processing

content	cleaning
Aplikasi ini sangat berguna untuk para pelajar maupun mahasiswa karena membantu menjawab pertanyaan dan bisa membantu hal lain, keren dah pokoknya	Aplikasi ini sangat berguna untuk para pelajar maupun mahasiswa karena membantu menjawab pertanyaan dan bisa membantu hal lain keren dah pokoknya

2.2.1. Text Pre-processing

The text processing process in text data analysis involves several key steps, including case folding, tokenization, stopword removal, normalization, and stemming. Case folding is converting all letters in the text to lowercase to standardize the text format, ensuring that the model or algorithm does not distinguish between uppercase and lowercase letters. Next, tokenization breaks the text into smaller units, called tokens, typically words or phrases to be used as units for analysis. Stopword removal eliminates words deemed insignificant, such as common conjunctions ("dan," "di," and "yang"), which frequently appear but do not provide meaningful information in the analysis. Normalization aims to standardize variations in word spellings, for instance, converting "tehnik" to "teknik" or "gmn" to "bagaimana," ensuring greater consistency in the text. Finally, stemming reduces words to their base or root form, such as transforming "kesalahan" and "salahan" into "salah," simplifying the text and reducing analysis complexity. These steps are crucial in preparing text for further analysis, such as sentiment classification, as they enhance the quality and consistency of the data used.

Table 2: Dataset after text pre-processing

Phase	Output
content	Aplikasi ini sangat berguna untuk para pelajar maupun mahasiswa karena membantu menjawab pertanyaan dan bisa membantu hal lain, keren dah pokoknya
cleaning	Aplikasi ini sangat berguna untuk para pelajar maupun mahasiswa karena membantu menjawab pertanyaan dan bisa membantu hal lain keren dah pokoknya
case_folding	aplikasi ini sangat berguna untuk para pelajar maupun mahasiswa karena membantu menjawab pertanyaan dan bisa membantu hal lain keren dah pokoknya
tokenization	['aplikasi', 'ini', 'sangat', 'berguna', 'untuk', 'para', 'pelajar', 'maupun', 'mahasiswa', 'karena', 'membantu', 'menjawab', 'pertanyaan', 'dan', 'bisa', 'membantu', 'hal', 'lain', 'keren', 'dah', 'pokoknya']
stopword_removal	['aplikasi', 'berguna', 'pelajar', 'mahasiswa', 'membantu', 'membantu', 'keren', 'dah', 'pokoknya']
normalization	aplikasi berguna pelajar mahasiswa membantu membantu keren deh pokoknya
stemming	aplikasi guna ajar mahasiswa bantu bantu keren deh pokok

2.3. Labeling

After the data undergoes the preprocessing stage, the next step is labeling. Labeling involves assigning sentiment labels to each data point, classified as positive, neutral, or negative. This labeling is performed automatically using the IndoBERT model, which has been trained for sentiment analysis in the Indonesian language. This step aims to accurately classify the sentiment within the data, allowing the results to be used in subsequent stages.

Following the labeling process using the IndoBERT model, a labeled dataset is obtained, with each data point assigned to the relevant sentiment category.

Table 3: Dataset after labeling

Content	**label**
salah fungsi terkadang fitur fungsi fungsi update fitur baca nyaring baca teks kerja lambat update aplikasi lambat freeze sebab kode ganggu kerja aplikasi salah tampil bug bikin tampil aplikasi rapi teks baca tombol muncul	Negatif
aplikasi guna ajar mahasiswa bantu bantu keren deh pokok	Positif
bagus sihh apk nya jawab akurat tapi ngelag yhh wifi jaring aman saja tapi ai nya ga respon sama sekali	Netral

2.4. Tranformation

To process and analyze textual data effectively, several techniques are used to extract meaningful features and reduce the complexity of the dataset. In this context, the text data is converted into a numeric representation using TF-IDF (Term Frequency-Inverse Document Frequency). The next step involves splitting the dataset into two parts: 20% for training data and 80% for testing data. This split is designed to ensure that the model is trained on a smaller subset of the data to learn patterns. In contrast, the more significant portion of the data tests the model's ability to generalize to new, unseen data. After data splitting, the next step is to apply Principal Component Analysis (PCA) to the dataset.

2.4.1. Term Frequency-Inverse Document Frequency (TF-IDF)

Term Frequency-Inverse Document Frequency (TF-IDF) in the weighting process aims to generate a vector encompassing a wide range of terms, where each word is calculated as a feature with a specific weight. This process enables the model to recognize each word's importance within a particular document's context [10]. These weighted terms can then be input for machine learning models to perform sentiment classification.

```
Hasil TF-IDF untuk beberapa baris:
   aaaa  aaaaahhhkkkk  aamiin  aandi  abad  abadi  abadin  abai  abalabal  \
0   0.0           0.0     0.0    0.0   0.0    0.0     0.0   0.0       0.0
1   0.0           0.0     0.0    0.0   0.0    0.0     0.0   0.0       0.0
2   0.0           0.0     0.0    0.0   0.0    0.0     0.0   0.0       0.0
3   0.0           0.0     0.0    0.0   0.0    0.0     0.0   0.0       0.0
4   0.0           0.0     0.0    0.0   0.0    0.0     0.0   0.0       0.0

   abang  ...  ytta   yu  yuhuuu  yuk  zaja  zalfa  zaman  zeroone  zionis  \
0    0.0  ...   0.0  0.0     0.0  0.0   0.0    0.0    0.0      0.0     0.0
1    0.0  ...   0.0  0.0     0.0  0.0   0.0    0.0    0.0      0.0     0.0
2    0.0  ...   0.0  0.0     0.0  0.0   0.0    0.0    0.0      0.0     0.0
3    0.0  ...   0.0  0.0     0.0  0.0   0.0    0.0    0.0      0.0     0.0
4    0.0  ...   0.0  0.0     0.0  0.0   0.0    0.0    0.0      0.0     0.0
```

Fig. 2: Output TF-IDF

2.4.2. Principal Component Analysis (PCA)

Principal Component Analysis (PCA) is a method used to reduce the dimensionality of attributes in a dataset, resulting in transformed values that are significantly different from their original form [4]. In addition to reducing dimensionality, Principal Component Analysis (PCA) is also useful for evaluating relationships between variables in a dataset. PCA can identify whether these variables are highly correlated or entirely unrelated. By understanding these inter-variable relationships, PCA aids in uncovering the data structure and serves as a valuable tool for preparing data for further analysis [11].

```
# Apply PCA to reduce dimensions
pca = PCA(n_components=0.95, svd_solver='full')  #0.95 to maintain 95% variance
X_train_pca = pca.fit_transform(X_train.toarray()) # Convert to array before PCA
X_test_pca = pca.transform(X_test.toarray())
```

Fig. 3: PCA implementation

Principal Component Analysis (PCA) was applied to reduce the dimensionality of a dataset while preserving 95% of the total variance. The process involves transforming the data into a new basis consisting of linear combinations of the original features, ordered by decreasing variance, using `n_components=0.95` to ensure sufficient components are selected. The complete singular value decomposition (SVD) was computed using `svd_solver='full'`. Before applying PCA, feature datasets (`X_train` and `X_test`) were converted into numerical arrays using the `toarray()` method, as required by the algorithm. The `fit_transform` method was employed to obtain the reduced-dimensional training data (`X_train_pca`), while the test data (`X_test`) was projected onto the same PCA basis using the `transform` method, resulting in `X_test_pca .`This approach reduced the dataset's dimensionality while retaining essential information, improving computational efficiency without significant variance loss.

2.5. Data Mining

Data mining is a process that leverages statistical, mathematical, artificial intelligence, and machine learning techniques to extract and identify patterns or valuable information from available data [12]. In the Data Mining phase, the Support Vector Machine (SVM) algorithm is applied to classify the data based on patterns identified in the previous steps. This process aims to categorize the sentiment within the text data. In this case, a linear kernel is used with the SVM algorithm. The linear kernel is a mathematical function that computes the dot product between input features, which allows the SVM to create a linear decision boundary between the classes. This kernel is particularly effective when the data is nearly linearly separable. Using the linear kernel, the algorithm constructs a decision hyperplane that best separates the sentiment categories based on the transformed features. This method is computationally efficient and effective when the number of dimensions is relatively high, as it minimizes the complexity of the model while still providing accurate classifications.

```
# Initialize SVM model
model_no_pca = SVC(kernel='linear')

# Train model with training data
model_no_pca.fit(X_train, y_train)

# Make prediction on test data
y_pred_no_pca = model_no_pca.predict(X_test)
```

Fig. 4: Implementation SVM

To enhance performance, a combined model of PCA and SVM is employed. By using PCA for dimensionality reduction prior to applying SVM, the approach is expected to yield more accurate and efficient classification results.

```
# Initialize SVM model with linear kernel
model_with_pca = SVC(kernel='linear')

# Train model with training data
model_with_pca.fit(X_train_pca, y_train)

# Make prediction on test data
y_pred_with_pca = model_with_pca.predict(X_test_pca)
```

Fig. 5: Implementation PCA and SVM

2.6. Evaluation

The evaluation phase aims to assess the performance of the sentiment analysis model using the Support Vector Machine (SVM) algorithm, both with and without the application of Principal Component Analysis (PCA). Evaluation is performed using evaluation metrics such as precision, recall, F1-score, and accuracy, which quantify how well the model correctly classifies sentiment categories (positive, negative, or neutral).

The confusion matrix provides a detailed breakdown of the model's predictions. It compares the predicted labels with the true labels, showing true positives, false positives, true negatives, and false negatives. This matrix helps to evaluate the overall accuracy and the model's ability to differentiate between sentiment classes, highlighting areas where misclassifications may occur. The confusion matrix functions by comparing the predictions generated by the model with the actual labels of the data. Through this evaluation, the performance of the model can be assessed by calculating metrics such as accuracy, precision, and recall. In classification tasks, the results analyzed using the confusion matrix represent four possible outcomes, which indicate the correct and incorrect predictions made by the model. These outcomes provide insights into the model's performance and its ability to distinguish between classes [13].

3. Result and Discussion

In this study, sentiment analysis was performed using the IndoBERT model, labeled to classify text sentiment into three main categories: Neutral, Positive, and Negative.

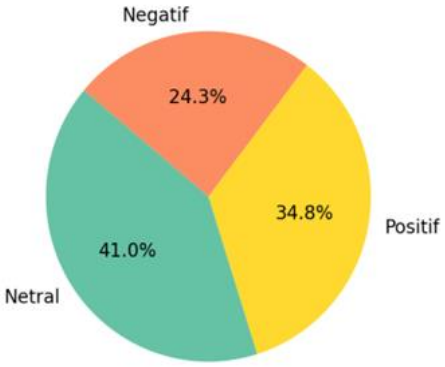

Fig. 6: Sentiment distribution

Table 4: Sentiment distirbution

Label	count
Netral	4097
Positif	3477
Negatif	2426

After the labeling process, it was found that the Neutral class dominated the dataset with 4097 entries, significantly outnumbering the Positive (3477) and Negative (2426) classes. This class imbalance indicates that most of the texts in the dataset did not exhibit strong or clear sentiments. Furthermore, there was a tendency for the model to classify texts as Neutral more frequently than as Positive or Negative, which could impact the model's overall performance. This imbalance may lead the model to classify texts as neutral more often, while positive and negative sentiments may not be optimally detected. Therefore, addressing the data imbalance issue is crucial, as well as implementing techniques such as class weighting or oversampling of the minority classes to improve the model's accuracy in detecting positive and negative sentiments.

Evaluation using a confusion matrix provides a detailed depiction of the number of correct and incorrect predictions for each category, making it a crucial tool for assessing the performance of classification models.

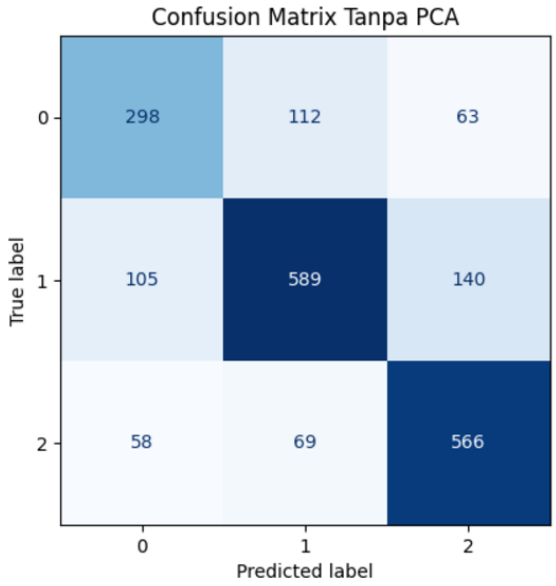

Fig. 7: Confusion Matrix SVM

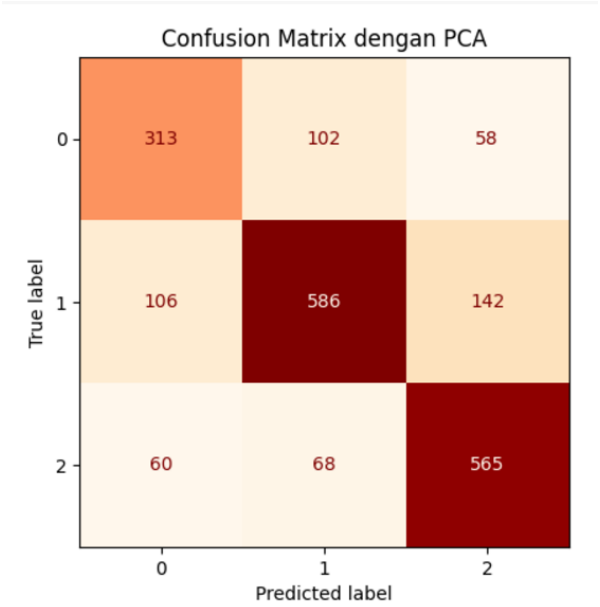

Fig. 8: Confusion Matrix PCA and SVM

Dimensionality reduction using Principal Component Analysis (PCA) did not significantly change the overall classification performance of the Support Vector Machine (SVM). Although a slight improvement in accuracy was observed in certain classes, such as label 0, dimensionality reduction with PCA showed a minor decrease in accuracy for other classes, such as label 1 and label 2. These results indicate that PCA can enhance computational efficiency without significantly compromising accuracy, mainly when applied to high-dimensional datasets. This underscores the potential of PCA as a supportive method for managing large-dimensional data in sentiment classification tasks.

The effect of Principal Component Analysis (PCA) on the performance of Support Vector Machine (SVM) in sentiment classification positively impacts model accuracy. In this study, applying PCA as a dimensionality reduction method enhanced sentiment classification performance. PCA plays a key role in reducing the dimensionality of the data without losing important information, thereby enabling SVM to focus on key features and reduce noise in the data, ultimately resulting in a more accurate and efficient model.

Table 5: Evaluation Metrix

Model	***Accuracy* (%)**	***Precision* (%)**	***Recall* (%)**	***F1-score* (%)**
SVM	72.65	71.58	71.77	71.56
SVM* dengan *PCA	73.20	72.24	72.66	72.32

Based on the results in the table above, the SVM model with and without PCA shows that PCA improves classification performance. The accuracy increased from 72.65% with the SVM model without PCA to 73.20% with PCA. Additionally, the precision value increased from 71.58% to 72.24%, indicating that the model with PCA is better at accurately identifying positive predictions. In terms of recall, there was an improvement from 71.77% to 72.66%, meaning the model with PCA was more effective at detecting all positive samples. Finally, the F1-score, which reflects the balance between precision and recall, also improved from 71.56% to 72.32%. Overall, applying PCA contributes to improving the performance of the SVM model by reducing the data dimensionality and enabling the identification of more relevant patterns.

These findings are consistent with previous research in the journal titled "Analisis Komparasi Metode Klasifikasi Data Mining dan Reduksi Atribut Pada Data Set Penyakit Jantung", which demonstrated that combining PCA with classification algorithms, such as Naïve Bayes, enhanced classification accuracy on complex medical data. Furthermore, the study indicated that applying PCA in the clustering process improved the clustering quality by eliminating less relevant features and preserving meaningful ones [14]. Similarly, these results align with another study published in the journal titled "Penerapan Principal Component Analysis (PCA) Untuk Reduksi Dimensi Pada Proses Clustering Data Produksi Pertanian Di Kabupaten Bojonegoro," which demonstrated that dimensionality reduction using PCA improves the quality of clustering models for agricultural production data using the K-Means algorithm. Both studies align in concluding that PCA, as a dimensionality reduction technique, is effective in reducing data complexity, thereby enabling machine learning algorithms to operate more efficiently and produce more accurate results [15].

4. Conclusion

The application of Principal Component Analysis (PCA) to the Support Vector Machine (SVM) model has been shown to improve performance in sentiment classification by simplifying high-dimensional datasets without losing important information. PCA helps the SVM focus on key features and reduces noise, making pattern detection more efficient.

Evaluation metrics indicate improvements in accuracy (from 72.65% to 73.20%), precision (from 71.58% to 72.24%), recall (from 71.77% to 72.66%), and F1-score (from 71.56% to 72.32%) following the application of PCA. Although the improvements are relatively small, the use of PCA has proven to enhance the efficiency and effectiveness of the SVM model in sentiment classification on complex datasets.

References

[1] K. A. Rokhman, B. Berlilana, and P. Arsi, "Perbandingan Metode Support Vector Machine Dan Decision Tree Untuk Analisis Sentimen Review Komentar Pada Aplikasi Transportasi Online," *J. Inf. Syst. Manag.*, vol. 3, no. 1, pp. 1–7, 2021, doi: 10.24076/joism.2021v3i1.341.

[2] C. F. Hasri and D. Alita, "Penerapan Metode NaãVe Bayes Classifier Dan Support Vector Machine Pada Analisis Sentimen Terhadap Dampak Virus Corona Di Twitter," *J. Inform. dan Rekayasa Perangkat Lunak*, vol. 3, no. 2, pp. 145–160, 2022, doi: 10.33365/jatika.v3i2.2026.

[3] Rayuwati, Husna Gemasih, and Irma Nizar, "Implementasi Algoritma NaiveBayes Untuk Memprediksi Tingkat Penyebaran Covid-19 Di Indonesia," *Jural Ris. Rumpun Ilmu Tek.*, vol. 1, no. 1, pp. 38–46, 2022, doi: 10.55606/jurritek.v1i1.127.

[4] D. A. Nugraha and A. S. Wiguna, "Seleksi Fitur Warna Citra Digital Biji Kopi Menggunakan Metode Principal Component Analysis," *Res. Comput. Inf. Syst. Technol. Manag.*, vol. 3, no. 1, p. 24, 2020, doi: 10.25273/research.v3i1.5352.

[5] M. H. Wicaksono, M. D. Purbolaksono, and S. Al Faraby, "Perbandingan Algoritma Machine Learning untuk Analisis Sentimen Berbasis Aspek pada Review Female Daily," *eProceedings Eng.*, vol. 10, no. 3, pp. 3591–3600, 2023.

[6] M. Septiani, "Pengenalan Pola Batik Lampung Menggunakan Metode Principal Component Analysis," *J. Inform. dan Rekayasa Perangkat Lunak*, vol. 2, no. 4, pp. 552–558, 2022, doi: 10.33365/jatika.v2i4.1612.

[7] I. A. Sapitri and M. Fikry, "Pengklasifikasian Sentimen Ulasan Aplikasi Whatsapp Pada Google Play Store Menggunakan Support Vector Machine," *J. TEKINKOM*, vol. 6, no. 1, pp. 1–7, 2023, doi: 10.37600/tekinkom.v6i1.773.

[8] P. Apriyani, A. R. Dikananda, and I. Ali, "Penerapan Algoritma K-Means dalam Klasterisasi Kasus Stunting Balita Desa Tegalwangi," *Hello World J. Ilmu Komput.*, vol. 2, no. 1, pp. 20–33, 2023, doi: 10.56211/helloworld.v2i1.230.

[9] M. D. Purbolaksono, M. Irvan Tantowi, A. Imam Hidayat, and A. Adiwijaya, "Perbandingan Support Vector Machine dan Modified Balanced Random Forest dalam Deteksi Pasien Penyakit Diabetes," *J. RESTI (Rekayasa Sist. dan Teknol. Informasi)*, vol. 5, no. 2, pp. 393–399, 2021, doi: 10.29207/resti.v5i2.3008.

[10] D. Darwis, E. S. Pratiwi, and A. F. O. Pasaribu, "Penerapan Algoritma Svm Untuk Analisis Sentimen Pada Data Twitter Komisi Pemberantasan Korupsi Republik Indonesia," *Edutic - Sci. J. Informatics Educ.*, vol. 7, no. 1, pp. 1–11, 2020, doi: 10.21107/edutic.v7i1.8779.

[11] Baiq Nurul Azmi, Arief Hermawan, and Donny Avianto, "Analisis Pengaruh Komposisi Data Training dan Data Testing pada Penggunaan PCA dan Algoritma Decision Tree untuk Klasifikasi Penderita Penyakit Liver," *JTIM J. Teknol. Inf. dan Multimed.*, vol. 4, no. 4, pp. 281–290, 2023, doi: 10.35746/jtim.v4i4.298.

[12] A. M. Argina, "Penerapan Metode Klasifikasi K-Nearest Neigbor pada Dataset Penderita Penyakit Diabetes," *Indones. J. Data Sci.*, vol. 1, no. 2, pp. 29–33, 2020, doi: 10.33096/ijodas.v1i2.11.

[13] E. Suryati, Styawati, and A. A. Aldino, "Analisis Sentimen Transportasi Online Menggunakan Ekstraksi Fitur Model Word2vec Text Embedding Dan Algoritma Support Vector Machine (SVM)," *J. Teknol. Dan Sist. Inf.*, vol. 4, no. 1, pp. 96–106, 2023, [Online]. Available: https://doi.org/10.33365/jtsi.v4i1.2445

[14] D. P. Utomo and Mesran, "Analisis Komparasi Metode Klasifikasi Data Mining dan Reduksi Atribut Pada Data Set Penyakit Jantung," *J. Media Inform. Budidarma*, vol. 4, no. 2, p. 437, 2020, doi: 10.30865/mib.v4i2.2080.

[15] D. Hediyati and I. M. Suartana, "Penerapan Principal Component Analysis (PCA) Untuk Reduksi Dimensi Pada Proses Clustering Data Produksi Pertanian Di Kabupaten Bojonegoro," vol. 05, pp. 49–54, 2021.