Sentiment-based tourism review classification using support vector machine with recursive feature elimination and synthetic minority oversampling

Husni[1*], *Mohammad* Syarief[1], *Andharini* Dwi Cahyani[1], *Ega* Herdianti Satria[1], *Ika* Oktavia Suzanti[1], *Aeri* Rachmad[1] and *Eka* Mala Sari Rochman[1]

[1]Departemen of Informatics, University of Trunodjoyo Madura, Bangkalan, Indonesia

Abstract. User-generated content, particularly sentiment-rich reviews, is a critical data source for the tourism industry. However, analyzing this data presents significant computational challenges: (1) high-dimensional feature spaces from text (e.g., TF-IDF) which increase computational complexity, and (2) severe class imbalance, which skews classifier performance towards the majority class (e.g., positive reviews). While deep learning models have emerged, classical machine learning offers robust alternatives when properly optimized. This study proposes and evaluates a hybrid classification framework based on Support Vector Machine (SVM), chosen for its proven efficacy in high-dimensional text classification. To address the aforementioned challenges, the model integrates two key techniques: Recursive Feature Elimination (RFE) to select the most salient features and reduce computational load, and the Synthetic Minority Oversampling Technique (SMOTE) to create a balanced data distribution for training. Experimental results demonstrate the profound impact of this integrated approach. Data balancing with SMOTE significantly improved model performance, boosting baseline accuracy from 76% to 96% (without RFE) and 78% to 95% (with RFE). The final optimized model (SVM+RFE+SMOTE) achieved high-performance metrics of 96% accuracy, 96% precision, 97% recall, and a 97% F1-score. Furthermore, RFE successfully reduced computation time (e.g., from 70s to 40s on a 956-review set). This study concludes that the proposed SVM+RFE+SMOTE framework is a highly effective and efficient method for sentiment-based classification of real-world tourism reviews.

1 Introduction

The tourism sector plays an important role in driving economic growth by generating income, creating jobs, and fostering inter-sectoral relationships. Various studies have shown that tourism contributes significantly to the growth of Gross Domestic Product (GDP) through foreign exchange earnings, job creation, and improving the quality of the regional economy [1], [2]. Therefore, the government and tourism destination managers must carry out promotion and improvement of destination quality in a sustainable manner. In addition, advances in technology that also present many social media present a lot of user content that automatically presents positive or negative opinions about a destination. Therefore, sentiment analysis of social media can also be used as a reference to present destinations that are of higher quality and have a positive image.

Sentiment analysis plays an important role in handling tourism destination reviews, supporting tourism organizations to identify user opinions and make informed decisions to improve services [3], [4], [5], [6], [7]. By analyzing opinions containing emotions and social media users' interests in certain tourism destinations through sentiment analysis, organizations can understand the emotional level associated with the destinations they manage, helping them to create strategic plans and marketing strategies to increase the attractiveness of the destination [4]. By utilizing sophisticated techniques such as the BERT model and attention fusion mechanisms (such as in Transformer), sentiment analysis can accurately extract emotional features from reviews, thereby improving the classification accuracy and effectiveness-efficiency of the decision-making process in the tourism sector [5]. In addition, sentiment analysis is also able to classify reviews into several different sentiment categories, such as positive, neutral, and negative. This facilitates easier decision-making, including in developing tourism organization management strategies [3], [7].

Recent studies have shown that the best machine learning methods for sentiment analysis of tourism reviews involve advanced deep learning techniques such as Long Short-Term Memory (LSTM) and Convolutional Neural Networks (CNN) [8]. These models have shown high levels of accuracy in sentiment analysis tasks, with LSTM achieving an accuracy of 96.42% on a hotel dataset. In addition, the use of Bidirectional LSTM (Bi-LSTM) and Convolutional Bi-LSTM models have shown effectiveness in sentiment analysis tasks, achieving accuracies of 86.09% and

* Corresponding author: husni@trunojoyo.ac.id

85.87%, respectively [2]. Furthermore, the application of Support Vector Machine (SVM) and K-Nearest Neighbor (KNN) algorithms has also shown promising results in classifying sentiment in tourism destination reviews, with SVM achieving a test accuracy of 95% and KNN achieving an accuracy of 84% [9]. These methods, when combined with attention mechanisms and fusion approaches, can improve the accuracy and efficiency of sentiment analysis in tourism reviews, aiding in the decision-making process for tourism organizations [3], [5], [10].

Support Vector Machine (SVM) shows several strengths for sentiment analysis, as highlighted in some research papers. SVM has been shown to outperform other algorithms such as Naive Bayes in terms of accuracy, with reported values ranging from 0.80394 to 0.93 [11], [12]. Additionally, SVM exhibits high accuracy, balanced precision, recall, and a robust ROC curve, making it effective in distinguishing between positive and negative sentiments in text classification tasks [13]. SVM has shown superior performance over Naive Bayes Classifier (NBC) in terms of accuracy, precision, recall, f-measure, and Area Under Curve (AUC), especially when dealing with imbalanced datasets [14]. SVM has also been successfully used in classifying review data with high percentages of accuracy, precision, and recall, making it a reliable choice for sentiment analysis applications [15]. These findings collectively emphasize the robustness and effectiveness of SVM in sentiment analysis tasks, making it a preferred choice for researchers and practitioners in the field.

SVM has several shortcomings, starting with the inability to handle the peculiarities of natural language in the form of ambiguous content such as typos and emoticons in reviews which can reduce the accuracy of the algorithm [3]. SVM also has difficulty in handling sarcasm in reviews which can lead to misinterpretation and affect the overall accuracy of sentiment analysis [16]. SVM may not always outperform other models in terms of precision, recall, and F1 score indicating that its performance can vary depending on the specific dataset and context [17]. Despite this, SVM remains a valuable tool in sentiment analysis, especially when combined with other techniques such as feature engineering to improve its accuracy and effectiveness [16]. Some other shortcomings of SVM are its high computational complexity on very large data (very large number of features), requires proper parameter tuning to optimize the performance of the model, cannot handle missing values, may not be effective for non-linearly separable data if not using kernel techniques, making SVM unsuitable for online or real-time applications.

2 Methods

The purpose of this study is to overcome the shortcomings of SVM, especially the high computational complexity by performing recursive feature selection. Several studies have shown that data class imbalance can limit the accuracy of SVM so that data balancing is needed, one of which is using oversampling. These two approaches are represented by the Recursive Feature Elimination (RFE) and Synthetic Minority Oversampling Technique (SMOTE) methods. The SVM Classifier is used to identify tourism destination reviews based on their sentiment into one of two classes: positive or negative. The combination of these 3 approaches is expected to improve the performance of the classifier in detecting review tendencies. Table 1 shows several articles that have combined SVM with RFE.

Table 1. Performance of SVM Classifier with RFE.

Article	Pre-processing + Feature Extraction	Methods	Result
[18]	Case folding Lemmatization Stop-word removal Tokenizing	SVM K-NN n-gram	SVM + Unigram is the best
[19]	Case Folding Tokenizing Stemming Normalization Stop-word Removal	SVM N-gram RFE Noise Reduction	SVM+RFE+Unigram is the best
[20]	Tokenizing stop-word Removal, Short-word removal, Stemming	SVM, RFE TF-IDF	High Accuracyshortest time
[21]	Cleaning Stop-word removal Stemming Tokenizing	SVM & BFTAN, SVM, BFTAN, TAN, NB, RF Word2Vec	SVM with feature selection is the best
[22]	Transform Case Tokenizing Normalization Stemming.	Naïve Bayes, SVM, k-NN	Highest accuracy of SVM

2.1 Architecture

Fig. 1 shows the architecture of the proposed approach to present a classifier with high accuracy and shorter computation time. In general, the system consists of:

1. **Input**. In this first stage, reviews of tourism destinations from Tripadvisor and Google Maps are collected. The precision of the keywords determines the quality and scope of the downloaded reviews. The keywords used are the names of the destinations, for example "Surabaya Zoo", "Asta Tinggi Sumenep", and "Pantai Sembilan". The downloaded reviews will be labeled manually and experts are asked to validate the labels.
2. **Text preprocessing**. This stage is very important in determining the quality of the data, including tokenizing (dividing sentences into a list of words, usually spaces and punctuation as separators between words), case folding (changing each letter in the review to lower case), cleaning (removing URLs, punctuation, numbers, hashtags, and usernames), normalization (changing non-standard or slang words into standard words with the help of

a slang word dictionary), filtering or stop-word removal (removing words that do not differentiate between reviews), and stemming or lemmatization (returning words to their basic form, usually applied to words with affixes) [23].

3. **Feature extraction** using TF-IDF (Term Frequency-Inverse Document Frequency).
4. **Data balancing**. In many cases, tourism destination datasets have unbalanced classes, sometimes the number of reviews in the positive class is much more than the number of reviews in the Negative class, sometimes vice versa. In this study, the number of reviews in the negative class is much less. The SMOTE technique is used to increase the number of reviews from the minority class (negative) so that the number is the same or balanced with the number in the Positive class. The generation of synthetic reviews in SMOTE uses a gamma value in the form of the distance between minority classes.
5. **Feature Selection and Modeling**. By default, big data has a very large number of features and this results in computational complexity. The RFE approach is used to recursively select the most important features to produce the best model with a minimum number of features. The dataset is divided into two, namely training and validation data.
6. **Testing** or Output. Testing data in the form of reviews is used to test the resulting model. The output of this final section is the sentiment class of the review: Positive or Negative.

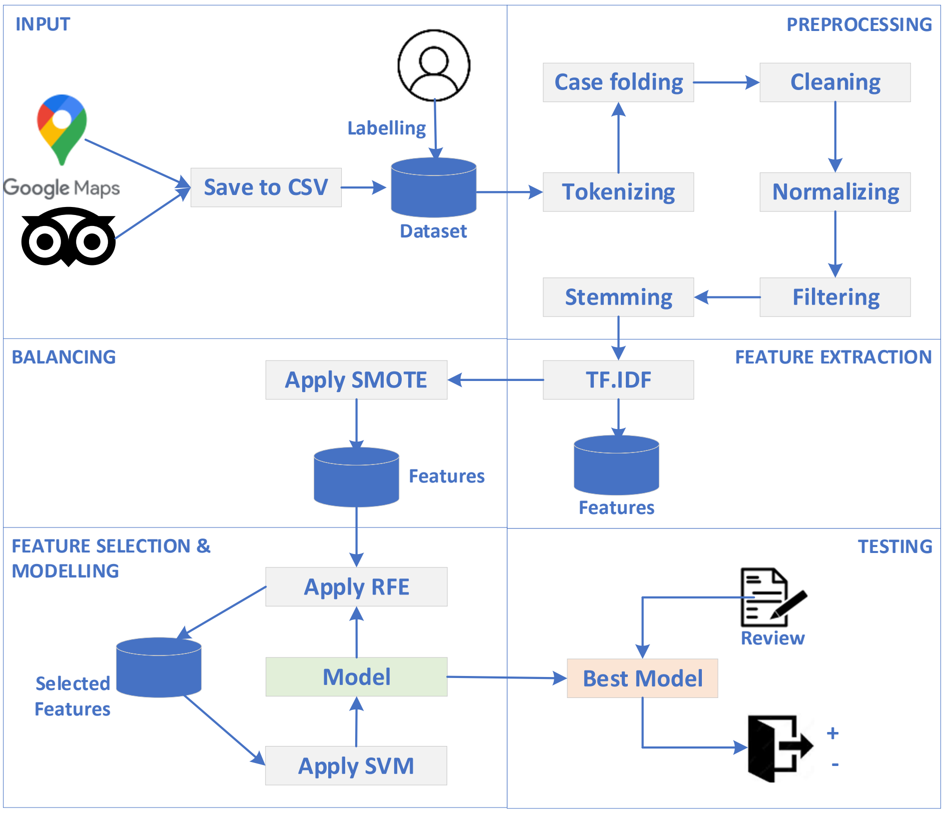

Fig. 1. Architecture of SVM-RFE with SMOTE.

2.2 Dataset dan preprocessing

Review data collection from Tripadvisor was done using a web scraping tool in Python called Selenium, while reviews from Google Maps can use the instance chrome extension. The total number of initial reviews collected was 760, of which 340 came from Tripadvisor. This dataset was labeled, validated and preprocessed. In text mining studies, words are termed as terms or features. Then term extraction or weighting was carried out following the TF-IDF formula (1), (2) and (3) [24].

$$TF = 0{,}5 + 0{,}5 \times \frac{TF}{max\,(TF)} \tag{1}$$

$$IDF_t = log\left(\frac{D}{DF_t}\right) \tag{2}$$

$$W_{d,t} = TF_{d,t} x\; IDF_{d,t} \tag{3}$$

Where:
TF = the number of occurrences of a word in a particular document or review
max(TF) = the highest number of occurrences of a particular term within the same review.
D = total number of reviews in the corpus
DF_t = document frequency, which is the number of reviews containing the term *t*.
IDF = inverse of *DF*.
D = document or review d^{th}
t = word, term, feature of the keyword
W = weight of d^{th} review against term *t*

The TF-IDF vector must be normalized with the Euclidean norm following the formula (4) [24]:

$$V_{norm} = \frac{v}{\sqrt{{v_1}^2 + {v_2}^2 + \cdots + {v_n}^2}} \tag{4}$$

In a searching system, a similarity measure indicates the level of relevance between two documents or a query and a document. A widely used approach is cosine similarity, which measures the cosine value (0 to 1) between two vectors in a vector space model. The cosine similarity between Query Q and Document D is shown in formula (5).

$$(Q, D) = \frac{Q \cdot D}{|Q||D|} \tag{5}$$

Where:
Q = TF-IDF of terms in Q
D = TF-IDF of terms in D
$|Q|$ = query length
$|D|$ = document length

Formula (5) will be used to measure, rank and eliminate features contained in the corpus when using RFE as a feature selection approach.

2.3 Data balancing: SMOTE

In text classification, imbalanced data can reduce the accuracy of the classifier. Data resampling techniques can be used to balance the data before the classifier is trained and the model is obtained. In general, there are two resampling approaches, namely: (1) Random Oversampling and (2) Random Under-sampling. Random Oversampling works by randomly generating more replicated samples from the minority label until there is a balance for all classes. This technique is also called naive random over-sampling. Oversampling can also create new samples by taking samples that have been randomly generated and replacing the original samples [25]. One of the Random Oversampling techniques is SMOTE [26]. This method is used to increase the amount of data from the minority class by replicating the data. The explanation above can be written with the formula (6).

$$dist = \sqrt{(X_1 + X_1)^2 + (X_2 + X_2)^2 + \cdots + (X_n + X_n)^2}$$
$$X_{syn} = X_i + (X_{knn} + X_1)^{\delta} \tag{6}$$

Where i is a positive integer number of keywords and δ is a random number between 0 and 1.

2.4 Feature selection: RFE

Recursive Feature Elimination (RFE) is a feature selection method for identifying key features in a dataset. It involves developing a model with the best features after repeatedly eliminating some of the least important features until the desired number of features is obtained. Although RFE can be used with any supervised learning method, Support Vector Machines (SVM) are the most popular pairing [27]. RFE ranks features using the formula (7):

$$W = \sum_{i=1}^{n} a_i y_i x_i \tag{7}$$

Where:
W = weight vector
x = attribute input value
a_i = support vector machine
y_i = data class

The RFE algorithm works in the following steps:

1. Rank the importance of each feature using the selected RFE machine learning algorithm.
2. Remove some of the least important features.
3. Build a model using the remaining (top-ranked) features.
4. Repeat steps 1-3 until the desired number of features is achieved.

2.5 Classifier: SVM

SVM is a supervised learning method that can be used for classification and regression tasks. In classification tasks, SVM has been mathematically mature and has a well-defined concept compared to other classification techniques. SVM can solve linear and nonlinear classification problems. SVM works by finding the optimal hyperplane to maximize the distance between classes [28]. Hyperplane is a feature that can be used to separate classes in a dataset as shown in Fig. 2.

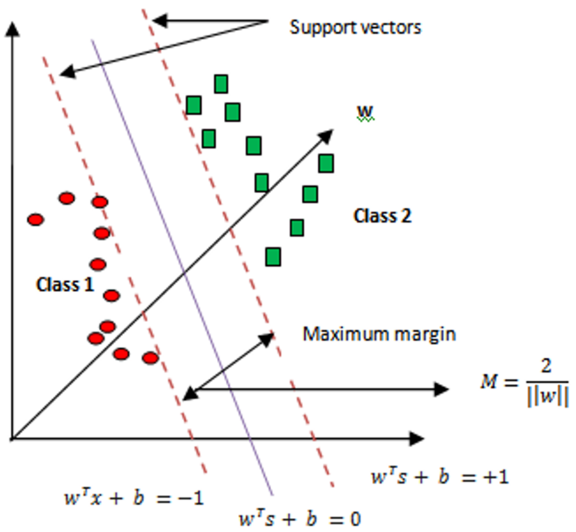

Fig. 2. Linear hyperplane in SVM.

According to [28] the SVM linear classification hyperplane is denoted in formula (8).

$$f(x) = w^t x + b \tag{8}$$

So we get equations (9) and (10).

$$[(w^t . xi) + b] \geq 1 \, for \, y1 = +1 \tag{9}$$
$$[(w^t . xi) + b] \leq -1 \, for \, y1 = -1 \tag{10}$$

Where x_i = training data set, $i = 1,2,3......n$.

The best hyperplane separates two class boundary planes maximally. This hyperplane is able to maximize the margin or distance between two sets of objects or reviews from different classes [28].

The margin value between 2 boundary planes based on the distance of the line to the center point can be represented by equation (11).

$$\frac{1-b-(-1-b)}{||W||}=\frac{2}{||W||} \tag{11}$$

This margin value is then maximized by minimizing the value of $||W||$ as the denominator.

2.6 Evaluation

There are 4 scenarios to assess the performance of the SVM classifier that has been built. Here are test scenarios to find out which approach is capable of delivering the best performance of SVM in sentiment analysis:

1. SVM. This test only uses SVM and involves all reviews, RFE and SMOTE are not applied.
2. SVM-SMOTE. This test balances the data class before running the SVM classifier as in point 1. RFE is not applied.
3. SVM-RFE. Unlike point 2, in this test RFE is applied but SMOTE is not.
4. SVM-RFE-SMOTE. In this test, the data class is balanced first using SMOTE before the feature selection RFE and training of the SVM classifier is carried out recursively.

Confusion matrix [29] is used to summarize and report performance in the form of accuracy, precision, recall and f1-score values from the classification process that has been carried out. The classification output label will be compared with the original label. The matrix for the two target classes (positive and negative) can be seen in Fig. 3, while the metrics used to measure the success of the classifier are formulas (12), (13), 14) and (15), namely accuracy, precision, recall, and F1 score. Each metric provides unique insights into model behavior, especially in scenarios with imbalanced data.

1. Accuracy measures the overall correctness of a model by calculating the ratio of true predictions to total predictions. However, it can be misleading in imbalanced datasets where the majority class dominates [30].
2. Precision or positive predictive value, indicates the proportion of true positive results among all positive predictions. It is crucial when the cost of false positives is high, such as in medical diagnoses [31].
3. Recall or sensitivity, measures the proportion of true positives identified by the model. It is particularly important in scenarios where missing a positive case (false negative) is critical [31].
4. The F1 score is the harmonic mean of precision and recall, providing a single metric that balances both concerns. It is especially useful in imbalanced datasets, where it can better reflect model performance than accuracy alone [30].

Figure 3 shows the concepts of true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN) which are very important for evaluating model performance. The differences between these four terms are summarized below:

1. True Positive (TP): Correctly identified positive instances. For example, in sentiment analysis, a correctly classified positive review is a TP [32].
2. True Negative (TN): Correctly identified negative instances. In the context of security testing, a correctly identified non-vulnerability is a TN [33].
3. False Positive (FP): Incorrectly identified positive instances. This occurs when a model predicts a positive outcome that is actually negative, such as misclassifying a benign review as positive [34].
4. False Negative (FN): Incorrectly identified negative instances. This is particularly problematic in sentiment analysis, where failing to identify a negative sentiment can lead to misleading conclusions [32].

		Actual Values	
		Positive	Negative
Predicted Values	Positive	True Positive (TP)	False Positive (FP)
	Negative	False Negative (FN)	True Negative (TN)

Fig. 3. Confusion Matrix with two target class.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN}\ X\ 100\% \tag{12}$$

$$Precision = \frac{TP}{TP+FP}\ X\ 100\% \tag{13}$$

$$Recall = \frac{TP}{TP+FN}\ X\ 100\% \tag{14}$$

$$F1\text{-}Score = 2x\,\frac{Recall\ x\ Precision}{Recall+\ Precision}\ X\ 100\% \tag{15}$$

Where:
TP = number of true positives (TP)
TN = number of true negatives (TN)
FP = number of false positives (FP)
FN = number of false negatives (FN)

3 Results

In the first scenario (only SVM), a classifier (model) is trained to classify reviews into one of the positive or negative sentiment class. Before being sent to the SVM, the dataset is divided into two: 50% is used as training data and the rest as validation data. The test records the best accuracy and the time required to achieve that accuracy value.

Fig. 4 shows the report provided by the scikit-learn library when running the SVM classifier with a linear kernel. The accuracy, precision, recall, and f1-score values of the SVM classifier on unbalanced data can be said to be sufficient (70% to 80%). The best accuracy of the SVM is 76%. The confusion matrix reflecting this report is shown in Fig. 5.

	precision	recall	f1-score	support
0	0.79	0.86	0.83	248
1	0.69	0.58	0.63	132
accuracy			0.76	380
macro avg	0.74	0.72	0.73	380
weighted avg	0.76	0.76	0.76	380

Fig. 4. Accuracy, precision, recall, and f1-score of SVM classifier without RFE on unbalanced data reported by scikit-learn.

		Actual Values	
		Positive	Negative
Predicted Values	Positive	TP=82	FP=30
	Negative	FN=59	TN=209

Fig. 5. Confusion matrix of SVM classifier without RFE on imbalanced data**.**

Based on the confusion matrix in Fig. 5 which shows True Positive = 82, False Positive = 59, True Negative = 209, and False Negative = 30, using formula (12) we obtain:

$$Accuracy = \frac{82+209}{82+209+59+30} \times 100\% = 76\%$$

$$Precision = \frac{82}{82+30} \times 100\% = 73\%$$

$$Recall = \frac{82}{82+59} \times 100\% = 58\%$$

$$F1\text{-}Score = 2x \frac{58\ x\ 73}{58+73}\ X\ 100\% = 65\%$$

In the second scenario where the data is balanced using SMOTE before being fed to the SVM, a confusion matrix is obtained as in Fig. 6. In this approach, the best accuracy is 96% which shows a significant increase from the first scenario of 20%. This test proves that data balancing using SMOTE is able to improve the performance of the SVM classifier. The condition of data before and after SMOTE is applied can be seen in Table 2.

		Actual Values	
		Positive	Negative
Predicted Values	Positive	TP=130	FP=7
	Negative	FN=9	TN=234

Fig. 6. Confusion matrix of SVM classifier without RFE on balanced data (second testing scenario).

Table 2. Dataset before and after SMOTE.

	Number of Reviews	Number of Positive	Number of Negatives
Before	760	478	282
After	956	478	478

Fig. 6 shows the True Positive value = 130, False Positive = 9, True Negative = 234, and False Negative = 7 so that using formula (12) we obtain:

$$Accuracy = \frac{130+234}{130+9+7+234} x\ 100\% = 96\%$$

$$Precision = \frac{130}{130+7}\ X\ 100\% = 95\%$$

$$Recall = \frac{130}{130+9}\ X\ 100\% = 94\%$$

$$F1\text{-}Score = 2x \frac{94\ x\ 95}{94+95}\ X\ 100\% = 95\%$$

		Actual Values	
		Positive	Negative
Predicted Values	Positive	TP=86	FP=58
	Negative	FN=27	TN=209

Fig. 7. Confusion matrix of SVM-RFE classifier on imbalanced data.

Next, in the third scenario, the SVM Classifier is run recursively along with the RFE feature selection but the data is not balanced with SMOTE. This test shows that this SVM-RFE approach provides a maximum accuracy of 78% when the number of features involved is 1992 out of a total of 2106 features. When compared to the first scenario, this test did not provide a significant increase in accuracy. The Confusion Matrix in Fig. 7 shows the magnitude of False Positive (58) and False Negative (27).

The final scenario is a test that runs SVM and recursive feature selection (SVM-RFE) on data that has been balanced with SMOTE. The result shows the classification metric values where the accuracy of SVM-RFE-SMOTE is able to provide the best accuracy of 95%. Fig. 8 shows that only a few reviews are misclassified (19), namely 12 False Positives and 7 False Negatives.

		Actual Values	
		Positive	Negative
Predicted Values	Positive	TP=134	FP=12
	Negative	FN=7	TN=227

Fig. 8. Confusion matrix of SVM-RFE classifier on balanced data (SMOTE).

The four test scenarios above show that the SVM classifier achieves the highest accuracy when all (unselected) terms are involved in model building on a dataset that has been balanced using SMOTE, which is 96%. Previously, when SVM was run on an unbalanced dataset, only 76% accuracy was obtained. This shows the great influence of dataset balancing in text classification. The first test (without SMOTE, without RFE) produced 76% accuracy and almost the same accuracy was given by the third test (RFE without SMOTE) which was 78%. These two tests show that feature selection is not able to significantly improve classifier accuracy. In essence, feature selection only selects and reduces the number of features so that the computation can be completed in a shorter time, not increasing accuracy. This can be seen from the reduction in computation time from 56 seconds to 25 seconds (Table 3) on the 760 reviews. This time difference will certainly be greater when the amount of data in the dataset is larger, for example above one million reviews.

Table 3. Summary of accuracy and computation time

#	Scenario	Best accuracy	Time (s)
1	SVM without RFE and SMOTE	76	56
2	SVM with SMOTE, No RFE	96	70
3	SVM with RFE, No SMOTE	78	25
4	SVM with RFE and SMOTE	95	40

The last scenario involving SMOTE and RFE is interesting to discuss because it is the focus of this study and is most likely to be the best approach. In this test, the accuracy obtained was 95%, almost the same as the second test (96%). The number of reviews that failed to be classified in these two tests was also almost the same, namely 16 (second test) compared to 19 (fourth test). These two tests emphasize the large influence of SMOTE oversampling on the performance of the SVM classifier while strengthening the previous analysis that feature selection including RFE is unable to improve the performance of the classifier. Feature selection is tasked with selecting the best features to reduce computation time, while data balancing functions to improve the quality of learning or training of the classifier. The fourth scenario provides an accuracy of 95%, almost the same as the second scenario but the time required to complete the computation is much smaller, namely 70 seconds (second scenario) compared to 40 seconds (fourth scenario) on a dataset containing 956 reviews.

4 Conclusions

The experimental results from the four conducted scenarios validate this study's primary hypotheses. It was empirically demonstrated that applying feature selection and data balancing yields significant improvements in both computational efficiency and model performance. Specifically, the application of Recursive Feature Elimination (RFE) resulted in a substantial reduction in training latency. For a dataset of 760 reviews, computation time was reduced from 56 seconds to 25 seconds. A similar reduction from 70 seconds to 40 seconds was observed on a larger dataset of 956 reviews. Concurrently, the implementation of the Synthetic Minority Oversampling Technique (SMOTE) for data balancing provided a pronounced improvement in predictive accuracy. In the model configuration without RFE, accuracy surged from 76% to 96%. A similar, significant gain from 78% to 95% was observed even when RFE was applied. These findings collectively underscore the critical roles of RFE for optimizing efficiency and SMOTE for enhancing predictive power in sentiment classification tasks, especially those involving high-dimensional and imbalanced datasets.

Based on these findings, several avenues for future research are recommended to further enhance the model's performance and robustness:

1. Expansion of the Class Model: Incorporating a Neutral class to accommodate reviews that exhibit ambiguity or lack clear positive or negative sentiment. This would transition the model from binary to multi-class classification, better reflecting the complexity of real-world data.
2. Dataset Augmentation and Diversity: Expanding the dataset volume and sourcing reviews from a wider array of platforms (e.g., social media, travel blogs). A larger, more heterogeneous dataset would improve the model's generalization capabilities against diverse linguistic styles.
3. Advanced Preprocessing: Implementing more sophisticated preprocessing techniques to manage informal language. This includes the normalization of domain-specific slang (slang words), the semantic interpretation of emoticons, and methods for ambiguity resolution.
4. Model Optimization and Tuning: A comprehensive exploration of alternative SVM kernels (e.g., Polynomial, RBF) beyond the linear kernel. Furthermore, integrating systematic hyperparameter tuning (e.g., Grid Search, Bayesian Optimization) is crucial to fully optimize the classifier's performance.

This research was supported by the Institute for Research and Community Service (LPPM), University of Trunodjoyo Madura, through the 2024 Research Group grant. The authors gratefully acknowledge this support.

References

1. N. Aida, G. Atiqasani, and W.A. Palupi, The Effect of the Tourism Sector on Economic Growth in Indonesia, WSEAS Transactions on Business and Economics, **(**2024). https://doi.org/10.37394/23207.2024.21.95
2. M.M. Uula, S. Maulida, and A.S. Rusydiana, Tourism Sector Development and Economic Growth in OIC Countries, Halal Tourism and Pilgrimage, **3**, 1, (2023). https://doi.org/10.58968/htp.v3i1.343
3. V.A. Narayana, B. Pooja, and B.A. Goud, Sentiment Analysis Based on Travelers' Reviews Using the SVM Model with Enhanced Conjunction Rule-Based Approach, IJRASET, **12**, 3, 1495–1502 (2024). https://doi.org/10.22214/ijraset.2024.59108
4. A.O. Çiğdemli, S. Yayla, and B.D. Çiğdemli, The world from the perspective of digital nomads: exploring sentiments in destination reviews, Worldwide Hospitality and Tourism Themes, **16**, 3, 303–315 (2024). https://doi.org/10.1108/whatt-03-2024-0044
5. D. Zhu, R. Jing, Q. Guo, D. Zhang, and F. Wan, Sentiment analysis of tourism review text combined with bert-bilstm and attention mechanism, Journal of Computational Methods in Sciences and Engineering, **24**, 3, 1605–1615 (2024). https://doi.org/10.3233/jcm-247135

6. H. Helen and R. Kurniawan, Sentiment Analysis of The Tourist Destination Using Support Vector Machine Algorithm on Twitter Post, ICIMTech, (2023). https://doi.org/10.1109/icimtech59029.2023.10277866
7. N.A.K.M. Haris et al., Sentiment classification from reviews for tourism analytics, International Journal of Advances in Intelligent Informatics, **9**, 1, 108–120 (2023). https://doi.org/10.26555/ijain.v9i1.1077
8. F. Amali, H. Yigit, and Z.H. Kilimci, Sentiment Analysis of Hotel Reviews using Deep Learning Approaches, IEEE eStream, (2024). https://doi.org/10.1109/estream61684.2024.10542593
9. Muhammad, S. Defit, and G.W.N. Cahyo, Comparison of Machine Learning Text Classification for Intent Sentiment Analysis, ICITISEE, (2023). https://doi.org/10.1109/icitisee58992.2023.10405200
10. K. Zhang, S. Wang, and Y. Yu, A TBGAV-Based Image-Text Multimodal Sentiment Analysis Method for Tourism Reviews, IJITWE, **18**, 1, 17 (2024). https://doi.org/10.4018/ijitwe.334595
11. T.A. Khan et al., Sentiment Analysis using Support Vector Machine and Random Forest, Journal of Informatics and Web Engineering, **3**, 1, 67–75 (2024). https://doi.org/10.33093/jiwe.2024.3.1.5
12. A.B. Setiawan, Sentiment Analysis Siap Kerja Training Services with Naive Bayes and SVM, Jurnal Ketenagakerjaan, **19**, 1, 102–111 (2024). https://doi.org/10.47198/jnaker.v19i1.303
13. H. Jadia, Comparative Analysis Of Sentiment Analysis Techniques: SVM, Logistic Regression, And TF-IDF Feature Extraction, IRJMETS, **5**, 10, 1182–1187 (2023). https://doi.org/10.56726/irjmets45265
14. Y.A. Singgalen, Sentiment Classification of Robot Hotel Content using NBC and SVM Algorithm, JoSYC, **5**, 2, 442–453 (2024). https://doi.org/10.47065/josyc.v5i2.4924
15. P. Irhami, E. Darnila, and F. Fadlisyah, Independent Campus Student Exchange Sentiment Analysis Using SVM, Journal of Advanced Computer Knowledge and Algorithms, **1**, 2, (2024). https://doi.org/10.29103/jacka.v1i2.14902
16. Kumar et al., Sarcasm Detection Using SVM, Cognitive Informatics and Soft Computing, **375**, (2022). https://doi.org/10.1007/978-981-16-8763-1_24
17. R. Setiyawan and Z. Mustofa, Comparison of the performance of naive bayes and support vector machine in sirekap sentiment analysis with the lexicon-based approach, Journal of Soft Computing Exploration, **5**, 2, 122–132 (2024). https://doi.org/10.52465/joscex.v5i2.367
18. P. Rahayu, A. Fauzi, and J. Indra, Analisis Sentimen Terhadap Program Kampus Merdeka Menggunakan Naive Bayes Dan Support Vector Machine, JSON, **4**, 2, 25–38 (2022). https://doi.org/10.30865/json.v4i2.5381
19. E.F. Saraswita, D.P. Rini, and A. Abdiansah, Analisis Sentimen E-Wallet di Twitter Menggunakan Support Vector Machine dan Recursive Feature Elimination, MIB, **5**, 4, 1195 (2021). https://doi.org/10.30865/mib.v5i4.3118
20. N.S.M. Nafis and S. Awang, An Enhanced Hybrid Feature Selection Technique Using Term Frequency-Inverse Document Frequency and Support Vector Machine-Recursive Feature Elimination for Sentiment Classification, IEEE Access, **9**, 52177–52192 (2021). https://doi.org/10.1109/ACCESS.2021.3069001
21. S.M. Alhashmi et al., Using a Hybrid-Classification Method to Analyze Twitter Data during Critical Events, IEEE Access, **9**, A, 141023–141035 (2021). https://doi.org/10.1109/ACCESS.2021.3119063
22. J.W. Iskandar and Y. Nataliani, Perbandingan Naïve Bayes, SVM, dan k-NN untuk Analisis Sentimen Gadget Berbasis Aspek, J. RESTI, **5**, 6, 1120–1126 (2021). https://doi.org/10.29207/resti.v5i6.3588
23. H. Husni, A. Kifli, M.A. Syakur, A. Muntasa, E.M.S. Rachman, H. Fauzan, and A. Rachmad, Classification of tourism destination review texts based on sentiment using k-nearest neighbor with information gain feature selection, ICoC-GTNT, 2024. https://doi.org/10.1063/5.0222729
24. C.D. Manning, P. Raghavan, and H. Schütze, Introduction to Information Retrieval, Cambridge University Press, 2008. https://doi.org/10.1007/s10791-009-9096
25. G. Zhao et al., Proceedings of the 2nd international conference on management engineering, software engineering and service sciences, 188–191 (2018). https://doi.org/10.1145/3180374.3181354
26. H. Sain and S.W. Purnami, Combine Sampling Support Vector Machine for Imbalanced Data Classification, Procedia Computer Science, **72**, A, 59–66 (2015). https://doi.org/10.1016/j.procs.2015.12.105
27. R.I. Pratama, S.A. Latipah, and B.N. Sari, Optimasi Klasifikasi Curah Hujan Menggunakan Support Vector Machine (Svm) Dan Recursive Feature Elimination (Rfe), JIPI, **7**, 2, 314–324 (2022). https://doi.org/10.29100/jipi.v7i2.2675
28. Destitus, W. Wella, and S. Suryasari, Support Vector Machine VS Information Gain: Analisis Sentimen Cyberbullying di Twitter Indonesia, Ultim. InfoSys, **11**, 2, 107–111 (2020). https://doi.org/10.31937/si.v11i2.1740
29. R.A. Barro, I.D. Sulvianti, and F.M. Afendi, Penerapan Synthetic Minority Oversampling Technique (SMOTE) Terhadap Data Tidak Seimbang Pada Pembuatan Model Komposisi Jamu, J. Stat., **1**, 1, 1–6 (2013). https://doi.org/10.29244/xplore.v1i1.12424

30. S. Riyanto et al., Comparative Analysis using Various Performance Metrics in Imbalanced Data for Multi-class Text Classification, International Journal of Advanced Computer Science and Applications, IJACSA, **14**, 6 (2023). https://doi.org/10.14569/ijacsa.2023.01406116
31. K. Takahashi, K. Yamamoto, A. Kuchiba, T. Koyama, Confidence interval for micro-averaged F1 and macro-averaged F1 Scores. Applied Intelligence, **52**, 4961–4972 (2022). https://doi.org/10.1007/s10489-021-02635-5
32. Chris Aloysius and P. Tamil Selvan, Reduction of false negatives in multi-class sentiment analysis, BEEI, **12**, 2, 1209–1218 (2023). https://doi.org/10.11591/eei.v12i2.4682
33. M. Shah and H. Lad, Efficient Classification of True Positive and False Positive XSS and CSRF Vulnerabilities Reported by the Testing Tool, in Proceedings of Second International Conference on Computing, Communications, and Cyber-Security. Springer, (2021). https://doi.org/10.1007/978-981-16-0733-2_62
34. V. kakollu, R. Pulicherla, G S. Sowmy, K. N. Soujanya, Predictive Analysis using Convolution Network on Sentiment Analysis of Text Classification using Machine Learning. IT in Industry, **9**, 2 (2021). https://doi.org/10.17762/ITII.V9I2.349