Enhancing Sentiment Analysis Performance Using SMOTE and Majority Voting in Machine Learning Algorithms

Fadli Suandi[1], M. Khairul Anam*[2], Muhammad Bambang Firdaus[3], Sofiansyah Fadli[4], Lathifah Lathifah[5], Eva Yumami[2], Alfa Saleh[6], Ade Zulkarnain Hasibuan[2]

[1] Politeknik Negeri Batam, Batam 29461, Indonesia
[2]Universitas Samudra, Langsa 24416, Indonesia
[3]Universitas Mulawarman, Samarinda 75119, Indonesia
[4]STMIK Lombok, Lombok Tengah 83511, Indonesia
[5]Universitas Teknokrat Indonesia, Bandar Lampung 35123, Indonesia
[6]Politeknik Negeri Bengkalis, Bengkalis 28711, Indonesia

khairulanam@unsam.ac.id

Abstract. In the digital era, sentiment analysis on social media has become increasingly important in understanding public perception of various issues. However, one of the main challenges in sentiment analysis is the issue of data imbalance, where one class (such as positive sentiment) may significantly outnumber another (such as negative or neutral sentiment). This imbalance can lead to biased predictions in machine learning models, where the majority class is favored over the minority class. To address this, Synthetic Minority Over-sampling Technique (SMOTE) is used to artificially balance the dataset by creating synthetic samples from the minority class. SMOTE generates new instances by interpolating between existing minority instances, improving the distribution of the data and enhancing model performance. In this research, various machine learning algorithms are utilized to perform sentiment analysis on tweets collected with the hashtag "online learning". The SMOTE oversampling technique is applied and compared with models that do not use SMOTE. This research focuses mainly on the Majority Voting algorithm, which combines predictions from multiple models to improve overall accuracy. The test results show that using SMOTE significantly improves the model's performance, especially in terms of recall and F1-Score. The Majority Voting+SMOTE algorithm achieved the highest accuracy of 97%, demonstrating the effectiveness of this approach in handling data imbalance and producing more reliable predictions. These results confirm that SMOTE effectively improves model performance under imbalanced data conditions, especially in sentiment analysis.

Keywords: Machine Learning, Majority Voting, SMOTE, Sentiment Analysis

L. Lumombo et al. (eds.), *Proceedings of the 7th International Conference on Applied Engineering (ICAE 2024)*,
Advances in Engineering Research 251,
https://doi.org/10.2991/978-94-6463-620-8_10

1 Introduction

Sentiment analysis, also known as opinion mining, is a text analysis process to determine the emotions or sentiments contained in it, whether positive, negative, or neutral [1]. With the development of technology, sentiment analysis has become an important tool in various fields, especially in understanding public opinion regarding certain products, services, or issues [2]. The benefits of sentiment analysis are very broad, from helping companies understand how consumers receive their products or services to enabling deeper analysis of market trends and consumer preferences. In addition, reputation management also uses sentiment analysis to monitor customer feedback and respond to certain campaigns or initiatives more effectively [3].

In its application, machine learning is the main approach in sentiment analysis. Machine learning algorithms allow systems to learn patterns and characteristics of text data automatically to increase the accuracy of sentiment predictions [4]. However, although powerful, machine learning algorithms are not without weaknesses. One of the main areas for improvement is the algorithm's tendency to overfit, especially when dealing with imbalanced datasets or high data complexity [5]. Therefore, there needs to be a method to handle data imbalance, one of which is the SMOTE technique [6].

Previous research in sentiment analysis has shown various approaches and significant results. Research [7] shows that the use of classification models such as Naive Bayes (NB), K-Nearest Neighbors (KNN), and Support Vector Machines (SVM) can achieve quite good results in sentiment analysis tasks on Indonesian online travel agents. Furthermore, research [8] conducted experiments with various feature extractions using random forests on the Amazon dataset. From this study, it was found that TF-IDF obtained the highest accuracy. Another study [9] increased accuracy by using majority voting and the results can be significantly improved compared to using based algorithms such as NB, Multilayer Perceptron (MLP), Decision Tree (DT), and Logistic Regression (LR).

This study chose three main algorithms: KNN, Random Forest (RF), and Support Vector Machine (SVM). KNN was chosen because of its simplicity and ability to capture patterns in the dataset based on the proximity between data [9]. Despite its simplicity, KNN can often provide good results in classification tasks when applied to datasets with a balanced class distribution [10]. Random Forest was used because of its strong ability to handle large and complex datasets and reduce the risk of overfitting through ensemble techniques of multiple decision trees [11]. Random Forest is also known for its ability to handle data with many features and capture complex interactions between them [12]. SVM was chosen because of its high performance in various classification tasks, especially in the case of imbalanced data [13]. SVM works by finding a hyperplane that maximizes the margin between different classes, effectively detecting differences between different classes [14].

This research also employs SMOTE, which is used to address dataset issues such as data imbalance. In addition to balancing the data, SMOTE can also improve model performance, as demonstrated by several previous studies. In a study conducted by [15], before using SMOTE, the SVM model achieved only 72% accuracy, but after applying SMOTE, the accuracy increased to 82%. Another study also utilized SMOTE; prior to

its application, the Random Forest model achieved 81% accuracy. After applying SMOTE, the accuracy of the Random Forest model increased to 97% [16].

In addition, this study also uses the Majority Voting technique to combine the results of the three algorithms. Majority Voting was chosen because it reduces model variability and improves overall accuracy [17]. By combining results from multiple models, this approach can take advantage of the strengths of each algorithm while mitigating individual weaknesses. Majority Voting works by collecting predictions from each model and determining the final result based on the most votes, which often results in more accurate and reliable decisions than a single model [18].

This study applied the KNN, Random Forest, and SVM algorithms for sentiment analysis. Each algorithm was tested with and without using SMOTE to handle data imbalance. In addition, this study combined the three algorithms using the Majority Voting Technique to see if this combination improved the overall performance of the model. The results of this study are expected to provide new insights into the application of machine learning for sentiment analysis, especially in the context of imbalanced and complex datasets.

2 Method

The following is the methodology used as a guide to conducting this research.

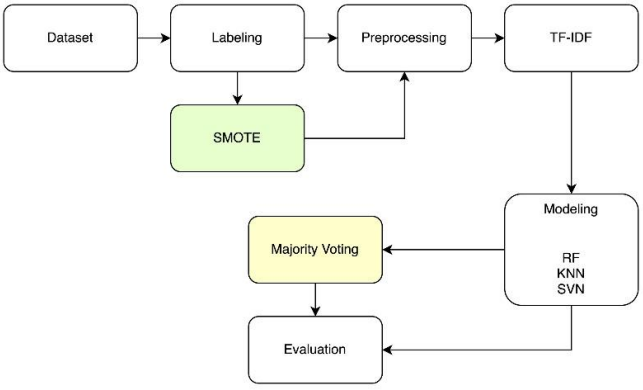

Fig. 1. Methodology Flow

2.1 Dataset

The dataset was collected from Twitter between January and June 2021 using the Drone Emprite Academic web platform. The data was not collected manually, but rather automatically by adding the hashtag #pembelajaran_daring to the platform. A total of 1200 tweets were collected for this analysis, and all tweets are in Indonesian. The dataset is divided into three categories: negative, neutral, and positive sentiment.

2.2 Labeling

In this research, sentiment analysis is performed by classifying the collected tweets

into three sentiment categories: positive, negative, and neutral. The labeling process is carried out based on specific parameters, which include the presence of certain keywords, context, and overall tone of the text.

- Positive: Tweets that contain supportive or favorable opinions, as well as optimistic expressions, are labeled as positive. These include tweets that use positive words such as "good," "excellent," or "happy," or tweets that express approval of online learning.
- Negative: Tweets that express disapproval, frustration, or dissatisfaction are labeled as negative. Negative sentiments are identified by the use of critical language, complaints, or negative expressions such as "bad," "difficult," or "frustrating."
- Neutral: Tweets that do not explicitly express strong positive or negative opinions are categorized as neutral. These include factual statements, questions, or comments that do not convey a clear emotional tone, such as "the online class started today" or "I have an assignment due tomorrow."

By using these parameters, each tweet is assigned a label that reflects the sentiment it expresses. This labeling process allows for a structured analysis of the overall sentiment distribution in the dataset, which is then used to train the machine learning models. Fig. 2 is the distribution of labels before data balancing.

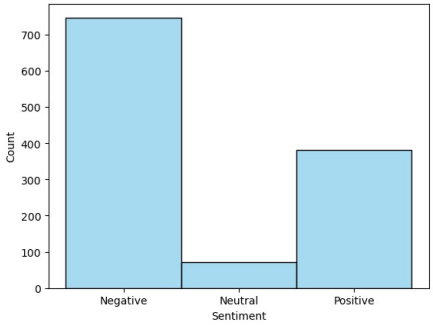

Fig. 2. Label Distribution

2.3 Synthetic Minority Over-Sampling Technique (SMOTE)

If the data in Fig. 2 is not balanced with the SMOTE oversampling technique, several serious problems can arise in training the machine learning model. First, the model tends to be biased towards the dominant class, in this case, the "negative" class, and tends to ignore underrepresented classes such as the "neutral" and "positive" classes. This can cause the model to be inaccurate in predicting the minority class.

In addition, models trained on imbalanced data will perform poorly in classifying the minority class. The model may predict the majority class more often due to the minority class's underrepresentation in the training data. This can be seen in evaluation metrics such as F1-score, precision, and recall, where the values for the minority class will be very low, indicating the model's inability to recognize examples from these classes.

Finally, models trained without oversampling are also at risk of overfitting the majority class, where the model becomes very accurate in predicting the majority class

but fails to generalize well to data from the minority class. Using SMOTE, examples from the minority class can be added to make the data more balanced. Allows the model to learn from more representative data and can improve the model's overall performance in classifying all classes more accurately. Fig. 3 is the label distribution after the class balancing process with SMOTE.

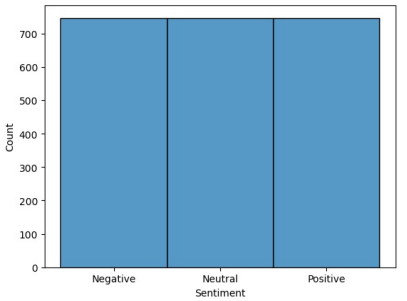

Fig. 3. Label distribution after class balancing with SMOTE.

2.4 Preprocessing

Preprocessing is an important step in processing text data, especially for data from Twitter, which often contains unstructured and diverse information [19]. In this project, preprocessing includes the following main steps:

- The first step in preprocessing is to remove columns that are irrelevant or do not provide information value in the analysis. For example, columns such as user_id, timestamp, or location may not be needed if the main focus is on the tweet text itself.
- Case Folding is the process of changing all letters in the text to lowercase. This is to ensure that words that should be considered the same but differ in capitalization (for example, "Learning" and "learning-ran") are treated consistently [20].
- Normalization involves the process of standardizing text by changing the form of words into a more common or standard format. For example, abbreviations, slang, or words that are often used on social media are changed into a more formal or standard form. This helps reduce the variation of words that actually have the same meaning.
- Tokenizing breaks down text into smaller units, usually words or tokens. For example, the sentence "Online learning is very effective" would be broken down into ["Learning," "online," "very," "effective"]. This tokenization is important for further analysis, such as word matching or weighting [21].
- Stopword removal is the process of removing common and frequently occurring words that do not contribute significantly to the context analysis, such as "and," "the, "or". Removing stopwords helps reduce the data's dimensionality and increases focus on more important words [22].
- Stemming is the process of reducing words to their basic form or "stem." For example, the words "belajar," "belajarannya," and "pembelian" would all be reduced to "belajar." This helps reduce variations in words that actually have the same basic meaning, thereby increasing the accuracy of text analysis [23].

This preprocessing process is very important to ensure that the text data used to train

the machine-learning model is clean, uniform, and ready for further analysis [23]. Each step above improves the data quality so that the resulting model can provide more accurate predictions.

2.5 TF-IDF

After preprocessing, the tweets are converted into a numeric representation using the TF-IDF (Term Frequency-Inverse Document Frequency) technique. This technique calculates the weight of each word in a tweet based on its frequency across tweets and gives a lower weight to words that frequently appear in all tweets (common words).

2.6 Modeling

After preprocessing, in the modeling stage, several machine learning algorithms are used to build a model that can classify tweets. The algorithms used include:

- Random Forest is an ensemble algorithm that combines many decision trees to make more stable and accurate predictions. One of its main advantages is its ability to handle imbalanced data and large numbers of features without easily overfitting. This algorithm also provides an estimate of the importance of each feature, which can help further understand the data [24].
- K-Nearest Neighbors (KNN) is a simple but effective non-parametric classification algorithm. Its main advantage is its ability to adapt to complex data without assuming a particular data distribution. This algorithm is easy to implement and often gives good results on small or medium datasets, especially when classification requires similarity or proximity-based decisions [25].
- Support Vector Machine (SVM) is very effective in high-dimensional spaces and continues to perform well even when the number of dimensions exceeds the number of samples. SVM uses a kernel function to handle non-linear cases by building a hyperplane that separates classes in the data with a maximum margin. Another advantage is that SVM tends to be more resistant to overfitting, especially on complex datasets with noise [26].

Utilizing various algorithms makes the modeling approach more robust because the advantages of each algorithm can be combined or used according to the characteristics of the data being analyzed. This approach allows for the production of models with optimal performance in various data conditions.

2.7 Majority Voting

After individual models are trained using the based algorithm, the majority voting technique combines predictions from various models. This technique allows for the final decision based on the majority of votes from all the different models.

2.8 Evaluation

The final stage is evaluation, where the resulting models are judged based on their performance in classifying tweets. Evaluation methods can include measurements such as accuracy, precision, recall, and F1-score. to ensure that the models perform well and are reliable.

3 Results and Discussion

3.1 Results

After the data preprocessing process, the next step is to test using the three algorithm-based models. Figure 4 is the accuracy result produced by the random forest algorithm without using SMOTE.

	precision	recall	f1-score	support
negatif	0.75	1.00	0.85	224
netral	1.00	0.05	0.09	21
positif	1.00	0.51	0.68	115
accuracy			0.79	360
macro avg	0.92	0.52	0.54	360
weighted avg	0.84	0.79	0.75	360

Fig. 4. Classification report random forest without SMOTE.

The evaluation results shown in the figure show the performance of the Random Forest algorithm applied without using the SMOTE technique to handle data imbalance. These results found that the precision value for the negative class was 0.75, meaning that 75% of all predictions categorized as negative by the model were negative classes. The precision is very high for the neutral and positive classes, each reaching 1.00, indicating that every prediction made for these classes is always correct.

However, the recall indicates that the model's performance between classes is very varied. The negative class has a very high recall, 1.00, which means that the model is able to correctly identify all negative instances. Conversely, the recall is very low for the neutral class, only 0.05, indicating that the model is almost unsuccessful in identifying neutral instances. The positive class has a recall of 0.51, indicating that only about half of the positive instances were successfully recognized by the model.

Then, the F1-score, the harmonic mean of precision and recall, shows that the overall performance also varies. The negative class has an F1-score of 0.85, indicating a good balance between precision and recall. However, the neutral class has a very low F1-score of 0.09, indicating that the model is ineffective in handling this class. The positive class has an F1-score of 0.68, indicating that despite high precision, lower recall reduces the model's effectiveness in detecting all positive instances.

The model's accuracy is 0.79, meaning about 79% of all model predictions are correct. However, the low recall and F1-score for the neutral and positive classes indicate that the model is more likely to be biased towards the majority (negative) class, indicating unaddressed data imbalance. One of the main problems when using a model without oversampling techniques, such as SMOTE, is designed to balance the class distribution and improve the model's ability to recognize all classes more fairly. Next is Fig. 5, which tests the random forest algorithm on a balanced dataset.

	precision	recall	f1-score	support
negatif	0.87	0.96	0.91	235
netral	1.00	0.98	0.99	214
positif	0.96	0.87	0.91	224
accuracy			0.93	673
macro avg	0.94	0.93	0.94	673
weighted avg	0.94	0.93	0.93	673

Fig. 5. Classification Report Random Forest with SMOTE

After using the SMOTE technique to handle data imbalance, the evaluation results of the Random Forest algorithm show a significant improvement compared to the model that does not use SMOTE. Regarding precision, the model with SMOTE increases the negative class from 0.75 to 0.87 and maintains high precision values in the neutral and positive classes, respectively, of 1.00 and 0.96. the model with SMOTE is more accurate in predicting the negative class and maintains a high level of accuracy in other classes.

In addition, recall, which was previously a major problem in the model without SMOTE, now shows a significant improvement. Recall for the neutral class has increased drastically from only 0.05 to 0.98, indicating that the model can now recognize almost all neutral instances correctly, likewise with the positive class, where recall increased from 0.51 to 0.87, indicating a marked improvement in detecting positive instances.

The increase in F1-score is also seen in all classes after using SMOTE. The F1-score for the negative and positive classes is now at 0.91, while for the neutral class, It increased sharply from 0.09 to 0.99. reflects that the model with SMOTE can achieve a much better balance between precision and recall than the model without SMOTE.

The model accuracy increased from 0.79 to 0.93 after applying SMOTE. This indicates that the model is better at classifying all classes fairly without being too biased toward the majority class. Using SMOTE has proven effective in overcoming data imbalance, resulting in a model with more reliable and accurate performance in predicting all classes. The following is a test of other algorithms presented in Table 1.

Table 1. Testing with other algorithms.

Algorithm	Accuracy	Precision	Recall	F1-Score
KNN	80%	55%	53%	53%
KNN+SMOTE	79%	84%	80%	78%
SVM	81%	89%	56%	57%
SVM+SMOTE	95%	95%	95%	95%
Majority Voting	80%	72%	65%	65%
Majority Voting+SMOTE	97%	97%	97%	97%

Based on the results of the Random Forest algorithm using SMOTE, we can compare the performance of this model with other algorithms listed in Table 1. Overall, Random Forest with SMOTE shows an accuracy of 93%, below the accuracy of SVM + SMOTE and Majority Voting + SMOTE, which reach 95% and 97%, respectively.

However, the performance of Random Forest remains quite competitive and shows a significant improvement compared to the model without SMOTE.

When viewed from the precision, recall, and F1-Score metrics, Random Forest with SMOTE shows a good balance, with precision and F1-Score values of 0.94 and recall of 0.93, respectively. These results are similar to those of SVM + SMOTE and Majority Voting + SMOTE, which have precision, recall, and F1-Score values of 0.95 and 0.97, respectively, indicating that this model successfully captures all classes well without bias towards certain classes.

Compared to the KNN algorithm, which has lower accuracy (80% without SMOTE and 79% with SMOTE), Random Forest with SMOTE is superior in accuracy and can recognize and classify all classes accurately. Although KNN + SMOTE shows a significant increase in precision and recall, namely 84% and 80%, the resulting F1-Score is still lower than Random Forest using SMOTE.

Overall, Random Forest with SMOTE positions itself as one of the strong models, especially in the context of imbalanced data. Although its performance is slightly below SVM + SMOTE and Majority Voting + SMOTE, this model still shows high reliability in balanced and fair classification across all classes.

3.2 Discussion

In this test, various machine learning algorithms are applied to classify tweets collected using the hashtag "online learning." This dataset has a significant imbalance between classes, which affects the model's classification performance. The SMOTE oversampling technique was applied, and a comparison was made between the models using SMOTE and those without.

The results of the tests showed that the use of SMOTE significantly improved the model performance, especially in the recall and F1-Score metrics, which are very important in the context of imbalanced data. For example, the Random Forest algorithm run without SMOTE showed quite good performance with an accuracy of 79% but had trouble recognizing the minority class, as seen from the low recall in the neutral and positive classes. After using SMOTE, the performance of Random Forest increased to 93% with more balanced recall and F1-Score values across all classes, indicating that this model is fairer in recognizing all classes.

Random Forest with SMOTE showed competitive performance compared to other algorithms also tested in this study, such as K-Nearest Neighbors (KNN) and Support Vector Machine (SVM). SVM and Majority Voting with SMOTE achieved the highest accuracies of 95% and 97%, respectively, indicating that they effectively handle data imbalance and produce highly accurate models. However, Random Forest with SMOTE remains in a strong position with 93% accuracy and an almost equally good balance of other metrics.

On the other hand, models without SMOTE generally showed weakness in recognizing minority classes, as seen from the low recall and F1-Score values for these classes. This highlights the importance of using techniques such as SMOTE in the context of imbalanced datasets to ensure that the model is accurate overall and fair in recognizing all classes.

This test confirms that the SMOTE technique is a very effective tool for improving the performance of machine learning models in imbalanced data conditions. While models such as SVM and Majority Voting performed the best, Random Forest with

SMOTE remains a strong and reliable choice, especially considering the performance balance across all metrics. SMOTE should be considered a standard step in imbalanced data preprocessing to maximize the effectiveness of machine learning models.

Overall, this study has demonstrated better performance improvements compared to previous studies, as shown in Table 2.

Table 2. Testing with other algorithms.

Researcher	Algorithm	Improvements Applied	Accuracy
[27]	C4.5	SMOTE	86%
[28]	Naïve Bayes	TF-IDF and SMOTE	89%
[26]	SVM	ADASYN	87,3%
[29]	Random Forest	SMOTE Tomek Links	86%
This Research	KNN, Random Forest, and SVM	Majority Voting + SMOTE	97%

Table 2 illustrates the comparison of various machine learning algorithms and techniques used in previous studies to improve performance in sentiment analysis. The results show that the application of SMOTE and other techniques such as TF-IDF, ADASYN, and Tomek Links yields accuracies ranging from 86% to 89% for algorithms such as C4.5, Naïve Bayes, SVM, and Random Forest. In contrast, the current study, which applies Majority Voting combined with SMOTE, achieves the highest accuracy of 97%, indicating a significant performance improvement compared to previous approaches.

4 Conclusion

This study shows that data imbalance is a significant problem that can affect the performance of machine learning models in text classification. By applying the SMOTE oversampling technique, the model performance, especially in terms of recall and F1-Score, can be substantially improved. The Random Forest algorithm with SMOTE achieved an accuracy of 93%, showing a significant improvement compared to the model without SMOTE. In addition, the SVM and Majority Voting algorithms with SMOTE also showed very good performance, achieving the highest accuracy in this test. In conclusion, SMOTE has proven to be a very effective technique in dealing with data imbalance and should be part of the preprocessing process in developing machine learning models to ensure more accurate and fair predictions.

Future research should explore other techniques that can work synergistically with SMOTE, such as a combination with an under-sampling algorithm or the application of more complex ensemble methods. In addition, it is important to evaluate the impact of SMOTE on different types of data, including very large data and data with more extreme levels of imbalance. Further research can also focus on developing and applying more sophisticated data augmentation techniques that can improve model performance without introducing bias or overfitting. Thus, machine learning models can be more adaptive and effective in various complex data conditions.

Acknowledgments. The authors would like to express their deepest gratitude to Politeknik Negeri Batam and Universitas Samudra for their support and facilities during this research. The assistance and cooperation provided by these two institutions were very meaningful in completing this research. We hope that the results of this research can positively contribute to the development of science and technology and be a useful reference in the future.

Disclosure of Interests. The authors declare that there are no conflicts of interest regarding the publication of this paper. The research was conducted independently and was not influenced by any commercial, financial, or institutional interests that could be construed as a potential conflict of interest. The authors are grateful to Politeknik Negeri Batam and Universitas Samudra for their support, but this support has not influenced the outcomes or interpretations of the research in any way.

References

1. M. K. Anam, S. Defit, Haviluddin, L. Efrizoni, and M. B. Firdaus, "Early Stopping on CNN-LSTM Development to Improve Classification Performance," *Journal of Applied Data Sciences*, vol. 5, no. 3, pp. 1175–1188, 2024, doi: 10.47738/jads.v5i3.312.
2. C. Atika Sari and E. Hari Rachmawanto, "Sentiment Analyst on Twitter Using the K-Nearest Neighbors (KNN) Algorithm Against Covid-19 Vaccination," *Journal of Applied Intelligent System*, vol. 7, no. 2, pp. 135–145, 2022, doi: 10.33633/jais.v7i2.6734.
3. H. Taherdoost and M. Madanchian, "Artificial Intelligence and Sentiment Analysis: A Review in Competitive Research," *Computers*, vol. 12, no. 2, pp. 1–15, Feb. 2023, doi: 10.3390/computers12020037.
4. A. Angdresey and G. Saroinsong, "The Decision Tree Algorithm on Sentiment Analysis: Russia and Ukraine War," *Jurnal Sisfotenika*, vol. 13, no. 2, pp. 192–200, 2023, doi: 10.30700/jst.v13i2.1397.
5. V. Werner de Vargas, J. A. Schneider Aranda, R. dos Santos Costa, P. R. da Silva Pereira, and J. L. Victória Barbosa, "Imbalanced data preprocessing techniques for machine learning: a systematic mapping study," *Knowl Inf Syst*, vol. 65, no. 1, pp. 31–57, Jan. 2023, doi: 10.1007/s10115-022-01772-8.
6. J. H. Joloudari, A. Marefat, M. A. Nematollahi, S. S. Oyelere, and S. Hussain, "Effective Class-Imbalance Learning Based on SMOTE and Convolutional Neural Networks," *Applied Sciences (Switzerland)*, vol. 13, no. 6, Mar. 2023, doi: 10.3390/app13064006.
7. A. D. Poernomo and S. Suharjito, "Indonesian online travel agent sentiment analysis using machine learning methods," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 14, no. 1, pp. 113–117, Apr. 2019, doi: 10.11591/ijeecs.v14.i1.pp113-117.
8. N. A. Semary, W. Ahmed, K. Amin, P. Pławiak, and M. Hammad, "Enhancing machine learning-based sentiment analysis through feature extraction techniques," *PLoS One*, vol. 19, no. 2, pp. 1–19, Feb. 2024, doi: 10.1371/journal.pone.0294968.
9. Z. Chen, L. J. Zhou, X. Da Li, J. N. Zhang, and W. J. Huo, "The Lao text classification method based on KNN," in *Procedia Computer Science*, Elsevier B.V., 2020, pp. 523–528. doi: 10.1016/j.procs.2020.02.053.
10. X. T. Dang and T. T. Le, "KNN-SMOTE: An Innovative Resampling Technique Enhancing the Efficacy of Imbalanced Biomedical Classification," in *Machine Learning and Other Soft Computing Techniques: Biomedical and Related Applications*, N. Hoang Phuong, N. T. Huyen Chau, and V. Kreinovich, Eds., Cham: Springer Nature Switzerland, 2024, pp. 111–121. doi: 10.1007/978-3-031-63929-6_11.
11. I. Setiawan *et al.*, "Utilizing Random Forest Algorithm for Sentiment Prediction Based on Twitter Data," in *Proceedings of the First Mandalika International Multi-Conference on*

Science and Engineering 2022, MIMSE 2022 (Informatics and Computer Science), Atlantis Press International BV, 2022, pp. 446–456. doi: 10.2991/978-94-6463-084-8_37.
12. L. K. Shrivastav and R. Kumar, "An Ensemble of Random Forest Gradient Boosting Machine and Deep Learning Methods for Stock Price Prediction," *Journal of Information Technology Research*, vol. 15, no. 1, pp. 1–19, Nov. 2021, doi: 10.4018/jitr.2022010102.
13. N. W. Susanto and H. Suparwito, "SVM-PSO Algorithm for Tweet Sentiment Analysis #BesokSenin," *Indonesian Journal of Information Systems (IJIS)*, vol. 6, no. 1, pp. 36–47, 2023, doi: 10.24002/ijis.v6i1.7551.
14. A. P. Gopi, R. N. S. Jyothi, V. L. Narayana, and K. S. Sandeep, "Classification of tweets data based on polarity using improved RBF kernel of SVM," *International Journal of Information Technology (Singapore)*, vol. 15, no. 2, pp. 965–980, Feb. 2023, doi: 10.1007/s41870-019-00409-4.
15. M. K. Anam, T. A. Fitri, Agustin, Lusiana, M. B. Firdaus, and A. T. Nurhuda, "Sentiment Analysis for Online Learning using The Lexicon-Based Method and The Support Vector Machine Algorithm," *ILKOM Jurnal Ilmiah*, vol. 15, no. 2, pp. 290–302, 2023, doi: 10.33096/ilkom.v15i2.1590.290-302.
16. W. Satria and M. Riasetiawan, "Essay Answer Classification with SMOTE Random Forest And Adaboost In Automated Essay Scoring," *IJCCS (Indonesian Journal of Computing and Cybernetics Systems)*, vol. 17, no. 4, p. 359, Oct. 2023, doi: 10.22146/ijccs.82548.
17. A. K. Abbas, A. K. Salih, H. A. Hussein, Q. M. Hussein, and S. A. Abdulwahhab, "Twitter Sentiment Analysis Using an Ensemble Majority Vote Classifier," *Journal of Southwest Jiaotong University*, vol. 55, no. 1, 2020, doi: 10.35741/issn.0258-2724.55.1.9.
18. S. Hadhri, M. Hadiji, and W. Labidi, "A voting ensemble classifier for stress detection," *Journal of Information and Telecommunication*, 2024, doi: 10.1080/24751839.2024.2306786.
19. K. Purwandari, T. W. Cenggoro, J. W. C. Sigalingging, and B. Pardamean, "Twitter-based classification for integrated source data of weather observations," *IAES International Journal of Artificial Intelligence*, vol. 12, no. 1, pp. 271–283, Mar. 2023, doi: 10.11591/ijai.v12.i1.pp271-283.
20. P. P. Putra, M. K. Anam, S. Defit, and A. Yunianta, "Enhancing the Decision Tree Algorithm to Improve Performance Across Various Datasets," *INTENSIF: Jurnal Ilmiah Penelitian dan Penerapan Teknologi Sistem Informasi*, vol. 8, no. 2, pp. 200–212, Aug. 2024, doi: 10.29407/intensif.v8i2.22280.
21. R. S. Putra, W. Agustin, M. K. Anam, L. Lusiana, and S. Yaakub, "The Application of Naïve Bayes Classifier Based Feature Selection on Analysis of Online Learning Sentiment in Online Media," *Jurnal Transformatika*, vol. 20, no. 1, pp. 44–56, Jul. 2022, doi: 10.26623/transformatika.v20i1.5144.
22. L. K. Ramasamy, S. Kadry, Y. Nam, and M. N. Meqdad, "Performance analysis of sentiments in Twitter dataset using SVM models," *International Journal of Electrical and Computer Engineering*, vol. 11, no. 3, pp. 2275–2284, Jun. 2021, doi: 10.11591/ijece.v11i3.pp2275-2284.
23. P. H. Prastyo, I. Ardiyanto, and R. Hidayat, "Indonesian Sentiment Analysis: An Experimental Study of Four Kernel Functions on SVM Algorithm with TF-IDF," in *International Conference on Data Analytics for Business and Industry: Way Towards a Sustainable Economy (ICDABI)*, Institute of Electrical and Electronics Engineers Inc., Oct. 2020, pp. 1–6. doi: 10.1109/ICDABI51230.2020.9325685.
24. M. R. Nugraha, M. D. Purbolaksono, and W. Astuti, "Sentiment Analysis on Movie Review from Rotten Tomatoes Using Modified Balanced Random Forest Method and Word2Vec," *Building of Informatics, Technology and Science (BITS)*, vol. 5, no. 1, pp. 153–161, Jun. 2023, doi: 10.47065/bits.v5i1.3596.
25. K. Satyanarayana, D. Shankar, and D. Raju, "An Approach For Finding Emotions Using Seed Dataset With Knn Classifier," *Turkish Journal of Computer and Mathematics Education*, vol.

12, no. 10, pp. 2838–2846, 2021, doi: 10.17762/turcomat.v12i10.4930.
26. N. G. Ramadhan, “Comparative Analysis of ADASYN-SVM and SMOTE-SVM Methods on the Detection of Type 2 Diabetes Mellitus,” *Scientific Journal of Informatics*, vol. 8, no. 2, pp. 276–282, Nov. 2021, doi: 10.15294/sji.v8i2.32484.
27. W. Rahayu *et al.*, “Synthetic Minority Oversampling Technique (SMOTE) for Boosting the Accuracy of C4.5 Algorithm Model,” *Journal of Artificial Intelligence and Engineering Applications*, vol. 3, no. 3, pp. 2808–4519, Jun. 2024, doi: 10.59934/jaiea.v3i3.469.
28. I. G. B. A. Budaya and I. K. P. Suniantara, “Comparison of Sentiment Analysis Algorithms with SMOTE Oversampling and TF-IDF Implementation on Google Reviews for Public Health Centers,” *MALCOM: Indonesian Journal of Machine Learning and Computer Science*, vol. 4, no. 3, pp. 1077–1086, Jul. 2024, doi: 10.57152/malcom.v4i3.1459.
29. H. Hairani, A. Anggrawan, and D. Priyanto, “Improvement Performance of the Random Forest Method on Unbalanced Diabetes Data Classification Using Smote-Tomek Link,” *International Journal on Informatics Visualization*, vol. 1, no. 7, pp. 258–264, Mar. 2023, doi: 10.30630/joiv.7.1.1069.