Sentiment Analysis of TikTok User Reviews on the Google Play Store

Dita Anggraini Soekarno[1*], Dewi Retno Sari Saputro[2], Sutanto[3]
[1,2,3]Departement of Mathematics and Natural Sciences, Universitas Sebelas Maret
Surakarta, 57126, Indonesia

Corresponding Author.
**Email*: ditaanggrni054@student.uns.ac.id

Abstract: *This study aims to analyze the sentiment of user reviews of the TikTok application on the Google Play Store to understand user perceptions of the services provided. The high usage of TikTok generates a large number of user reviews, making manual analysis inefficient; therefore, sentiment analysis is required to automatically identify user opinions. A total of 17,016 reviews were collected using web scraping techniques with the help of the* google-play-scraper *library in Python. The reviews underwent preprocessing stages, including cleaning, case folding, tokenizing, stopword removal, and stemming, followed by sentiment labeling using the Valence Aware Dictionary and sEntiment Reasoner (VADER). Text representation was transformed into numerical form using the Term Frequency–Inverse Document Frequency (TF-IDF) method, with an 80:20 split for training and testing data. To address data imbalance, the Adaptive Synthetic Sampling (ADASYN) method was applied, while classification was performed using the Random Forest algorithm. Evaluation based on the confusion matrix showed an accuracy of 92.39%, precision of 96.21%, recall of 92.47%, and F1-score of 94.30%. These results indicate that the model effectively classifies user review sentiments and provides insights into user perceptions of the TikTok application.*
Keywords: *Sentiment Analysis, TikTok Reviews, TF-IDF, Random Forest, ADASYN*

1. INTRODUCTION

The rapid development of information and communication technology has driven the significant growth of social media as a medium for interaction and information sharing. Social media is not only used for entertainment, but also serves as a platform for users to express opinions, share experiences, and provide evaluations of a service or product [1]. The information generated from these user interactions can be utilized to better understand user needs and satisfaction. One of the social media platforms that has experienced a significant increase in usage is TikTok, which offers short-video sharing services with various interactive features that are easily accessible to a wide range of users [2].

The large number of TikTok users has led to an increase in the volume of user reviews on application distribution platforms such as the Google Play Store. These reviews contain various opinions that reflect user experiences with the application, both in terms of satisfaction and dissatisfaction [3]. The information contained in user reviews is valuable for evaluation purposes in the development and improvement of application service quality. However, the large volume of review data makes manual analysis inefficient, time-consuming, and potentially prone to subjective bias in evaluation.

To address this issue, sentiment analysis is employed as an approach to automatically process textual data. Sentiment analysis is a part of text mining that aims to identify, extract, and classify user opinions into specific categories, such as positive, negative, or neutral [4]. By applying sentiment analysis, unstructured review data can be transformed into more organized information, making it easier to understand trends in user perceptions of an application.

In the sentiment analysis process, the preprocessing stage is a crucial step to clean and prepare the data before further analysis is conducted. Subsequently, text representation is performed using the Term Frequency–Inverse Document Frequency (TF-IDF) method, which assigns weights to words based on their importance within a document [5]. This method is widely used due to its simplicity and its ability to improve the quality of feature representation in textual data. Furthermore, the classification process is carried out using the Random Forest algorithm, which is known for its ability to handle high-dimensional data and its stable performance across various classification tasks [6].

Another common issue in sentiment analysis is the imbalance in the number of data points across sentiment classes. This condition can affect classification results, as the model tends to be biased toward the majority class. Therefore, data balancing techniques are required to address this problem. One method that can be applied is Adaptive Synthetic Sampling (ADASYN), which generates synthetic data for the minority class to achieve a more balanced data distribution [7]. Although numerous studies have explored sentiment analysis using various methods, the application of a combined approach involving TF-IDF, ADASYN, and Random Forest on TikTok application reviews remains limited, thus providing added value to this research in examining such a methodological combination.

Based on the above discussion, this study aims to analyze the sentiment of TikTok user reviews on the Google Play Store using the TF-IDF, ADASYN, and Random Forest methods. This research is expected to provide insights into user perceptions of the application and serve as a basis for evaluating and improving the quality of the services provided. Studies specifically focusing on sentiment analysis of TikTok user reviews are still limited, which highlights the relevance of this research.

2. METHOD

2.1. Data Source

The data used in this study consist of user reviews of the TikTok application obtained from the Google Play Store. Data collection was carried out using web scraping techniques with the assistance of the *google-play-scraper* library in the Python programming language. This process was performed automatically to extract information in the form of review text, ratings, and review dates. Web scraping enables the efficient collection of large amounts of data, thereby supporting a representative data-driven sentiment analysis [2].

2.2. Research Steps

The stages of this sentiment analysis study were carried out systematically as follows:

1. Data collection of user reviews using web scraping techniques.
2. Text data preprocessing to clean and prepare the data.
3. Sentiment labeling using the VADER method.
4. Text representation into numerical form using TF-IDF.
5. Data balancing using ADASYN.
6. Classification using the Random Forest algorithm.
7. Model evaluation using a confusion matrix and evaluation metrics.

2.3. Sentiment Analysis

Sentiment analysis is a part of text mining that aims to identify, extract, and classify opinions in textual data into specific categories, such as positive, negative, or neutral [1], [24]. This technique is widely used to analyze unstructured data from social media in order to understand user perceptions of a service or product [23].

In this study, sentiment analysis is focused on binary classification, namely positive and negative sentiments. This approach is used to simplify the classification process and improve the model's performance in distinguishing sentiment polarity more clearly.

2.4. Web Scraping

Web scraping is a technique for automatically collecting data from web pages by extracting relevant information [2]. In this study, web scraping is used to collect a large number of TikTok user reviews from the Google Play Store. The collected data are then stored and used as the main dataset in the sentiment analysis process.

2.5. Preprocessing

Text data obtained from the scraping process generally still contain noise such as punctuation, symbols, and variations in writing. Therefore, a preprocessing stage is carried out to clean and prepare the data before further analysis is conducted [3]. The preprocessing steps include:

1) Cleaning, which involves removing irrelevant characters such as punctuation, numbers, and symbols.
2) Case folding, which involves converting all letters to lowercase to standardize the text format.

3) Tokenizing, which involves splitting the text into individual word units.
4) Stopword removal, which involves removing common words that do not carry significant meaning.
5) Stemming, which involves reducing words to their base or root form.

These steps aim to reduce noise and improve the quality of the resulting features, thereby enhancing the performance of the classification model [3], [15].

2.6. VADER

Sentiment labeling is performed using the VADER (Valence Aware Dictionary and sEntiment Reasoner) method, which is a lexicon-based approach specifically designed for sentiment analysis of social media text [4].

VADER produces sentiment scores in the form of positive, negative, neutral, and a compound score. The compound score is a normalized value of the sentiment score ranging from -1 to 1, which is used to determine the sentiment polarity of a text. Based on this value, the text is then classified into either positive or negative categories.

2.7. TF-IDF

Text representation in this study is carried out using the Term Frequency–Inverse Document Frequency (TF-IDF) method, which is used to transform textual data into numerical representation based on the importance of a word within a document [5], [18].

Term Frequency (TF) is used to measure the frequency of a word's occurrence in a document, while Inverse Document Frequency (IDF) is used to measure the importance of a word across the entire collection of documents. The values of TF and IDF are formulated as follows:

$$TF(t,d) = \frac{f(t,d)}{\sum f(t,d)}$$

$$IDF(t) = log\left(\frac{N}{df(t)}\right)$$

The final weight of each word is obtained by multiplying the TF and IDF values as follows:

$$TF - IDF\ (t,d) = TF(t,d) \times IDF(t)$$

This method assigns higher weights to words that frequently appear in a particular document but rarely appear in other documents, thereby effectively representing important information in the sentiment classification process.

2.8. ADASYN

Data imbalance is a common issue in classification, where the number of instances across classes is not evenly distributed. This condition can cause the model to be biased toward the majority class [14], [17].

To address this issue, the Adaptive Synthetic Sampling (ADASYN) method is applied, which is an oversampling technique that generates synthetic data for the minority class [6]. ADASYN works adaptively by generating more synthetic data in regions that are harder for the model to learn, thereby making the data distribution more balanced and improving the learning process.

2.9. Random Forest

Random Forest is an ensemble learning–based classification algorithm that constructs multiple decision trees and combines the predictions from each tree to produce a final decision [7].

This algorithm employs bootstrap aggregating (bagging) and random feature selection at each node, which helps improve model accuracy and reduce the risk of overfitting [19]. Random Forest is also known to be effective in handling high-dimensional data such as text data [10].

2.10. Model Evaluation

The model evaluation was conducted using a confusion matrix to measure classification performance by comparing predicted results with actual data [8], [25]. The confusion matrix consists of four main components: true positive (TP), true negative (TN), false positive (FP), and false negative (FN). Based on these values, several evaluation metrics are used as follows:

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

$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}$$

$$F1 - Score = 2 \times \frac{Presisi \times Recall}{Presisi + Recall}$$

These metrics are used to comprehensively evaluate the model's performance in classifying user review sentiments.

3. RESULTS AND DISCUSSION

In this section, the results and discussion of the sentiment analysis process on user reviews of the TikTok application on the Google Play Store are presented. The discussion is divided into several stages, namely web scraping, preprocessing, sentiment labeling, text representation into numerical form, class balancing using ADASYN, the implementation of the Random Forest algorithm, and model evaluation.

3.1 Web Scraping

The web scraping process resulted in 17,016 user reviews, which were used as the dataset in this study. The collected data reflect a wide range of user experiences, including appreciation for the application's features as well as complaints related to performance and advertisements.

Table 1. The first two reviews obtained from the web scraping process

No	Review
1	So I love the app and I have for a while. However the new interruption ads suck. i'm switching back to red box they interrupt every video and I hate interruptions more than anything, at least wait till the end of the video, and before it restarts play it. but no, every single one i've gotten has been directly in the center of the thing, and i've closed it every time i'm done with tiktok, if this does not get fixed.
2	It has shorts and live, so its streaming content. But it also has an entire shop. It seem like a market advertising its primary service as jesters and other hecklers in the street. Kind of like an online blog. I have had some errors where the audio from a previous tiktok will play over my next, even going as far as to still play even when I leave the app. Some tiktoks will also not load until I go back and swipe back into them. I am also displeased in the direction its going with advertising.

The relatively large amount of data indicates that user reviews have high informational potential for analysis. The variety of opinions that emerge also suggests that the dataset is sufficiently representative in describing user perceptions of the TikTok application.

3.2 Preprocessing

The preprocessing results show that the text data underwent significant normalization. Irrelevant characters were successfully removed, all text was converted to lowercase, and words were tokenized and reduced to their base forms.

Table 2. The first two reviews after preprocessing

No	Preprocessing Stage	First review	Second review
1	Cleaning	So I love the app and I have for a while However the new interruption ads suck im switching back to red box they interrupt every video and I hate interruptions more than anything at least wait till the end of the video and before it restarts play it but no every single one ive gotten has been directly in the center of the thing and ive closed it every time im done with tiktok if this does not get fixed	It has shorts and live so its streaming content But it also has an entire shop It seem like a market advertising its primary service as jesters and other hecklers in the street Kind of like an online blog I have had some errors where the audio from a previous tiktok will play over my next even going as far as to still play even when leave the app Some tiktoks will also not load until I go back and swipe back into them I am also displeased in the direction its going with advertising
2	Case Folding	so i love the app and i have for a while however the new interruption ads suck im switching back to red box they interrupt every video and i hate interruptions more than anything at least wait till the end of the video and before it restarts play it but no every	it has shorts and live so its streaming content but it also has an entire shop it seem like a market advertising its primary service as jesters and other hecklers in the street kind of like an online blog i have had some errors where the audio from a previous tiktok will play over my next

		single one ive gotten has been directly in the center of the thing and ive closed it every time im done with tiktok if this does not get fixed	even going as far as to still play even when I leave the app some tiktoks will also not load until i go back and swipe back into them i am also displeased in the direction its going with advertising
3	Tokenizing	'so', 'i', 'love', 'the', 'app', 'and', 'i', 'have', 'for', 'a', 'while', 'however', 'the', 'new', 'interruption', 'ads', 'suck', 'im', 'switching', 'back', 'to', 'red', 'box', 'they', 'interrupt', 'every', 'video', 'and', 'i', 'hate', 'interruptions', 'more', 'than', 'anything', 'at', 'least', 'wait', 'till', 'the', 'end', 'of', 'the', 'video', 'and', 'before', 'it', 'restarts', 'play', 'it', 'but', 'no', 'every', 'single', 'one', 'ive', 'gotten', 'has', 'been', 'directly', 'in', 'the', 'center', 'of', 'the', 'thing', 'and', 'ive', 'closed', 'it', 'every', 'time', 'im', 'done', 'with', 'tiktok', 'if', 'this', 'does', 'not', 'get', 'fixed'	'it', 'has', 'shorts', 'and', 'live', 'so', 'its', 'streaming', 'content', 'but', 'it', 'also', 'has', 'an', 'entire', 'shop', 'it', 'seem', 'like', 'a', 'market', 'advertising', 'its', 'primary', 'service', 'as', 'jesters', 'and', 'other', 'hecklers', 'in', 'the', 'street', 'kind', 'of', 'like', 'an', 'online', 'blog', 'i', 'have', 'had', 'some', 'errors', 'where', 'the', 'audio', 'from', 'a', 'previous', 'tiktok', 'will', 'play', 'over', 'my', 'next', 'even', 'going', 'as', 'far', 'as', 'to', 'still', 'play', 'even', 'when', 'i', 'leave', 'the', 'app', 'some', 'tiktoks', 'will', 'also', 'not', 'load', 'until', 'i', 'go', 'back', 'and', 'swipe', 'back', 'into', 'them', 'i', 'am', 'also', 'displeased', 'in', 'the', 'direction', 'its', 'going', 'with', 'advertising'
4	Stopword Removal	'love', 'app', 'however', 'new', 'interruption', 'ads', 'suck', 'im', 'switching', 'back', 'red', 'box', 'interrupt', 'every', 'video', 'hate', 'interruptions', 'anything', 'least', 'wait', 'till', 'end', 'video', 'restarts', 'play', 'every', 'single', 'one', 'ive', 'gotten', 'directly', 'center', 'thing', 'ive', 'closed', 'every', 'time', 'im', 'done', 'tiktok', 'get', 'fixed'	'shorts', 'live', 'streaming', 'content', 'also', 'entire', 'shop', 'seem', 'like', 'market', 'advertising', 'primary', 'service', 'jesters', 'hecklers', 'street', 'kind', 'like', 'online', 'blog', 'errors', 'audio', 'previous', 'tiktok', 'play', 'next', 'even', 'going', 'far', 'still', 'play', 'even', 'leave', 'app', 'tiktoks', 'also', 'load', 'go', 'back', 'swipe', 'back', 'also', 'displeased', 'direction', 'going', 'advertising'
5	Stemming	love app howev new interrupt ad suck im switch back red box interrupt everi video hate interrupt anyth least wait till end video restart play everi singl one ive gotten directli center thing ive close everi time im done tiktok get fix	short live stream content also entir shop seem like market advertis primari servic jester heckler street kind like onlin blog error audio previou tiktok play next even go far still play even leav app tiktok also load go back swipe back also displeas direct go advertis

This process produces more consistent and structured text, thereby reducing noise in the data. As a result, the quality of the features generated in the subsequent stage improves and can enhance the performance of the classification model.

3.3 Sentiment Labeling

Based on labeling using the VADER method, the sentiment distribution from a total of 17,016 reviews consists of 8,367 positive reviews, 4,729 neutral reviews, and 3,920 negative reviews. In this study, neutral reviews were not used in the classification stage; therefore, the subsequent analysis only utilized data with positive and negative sentiments.

Table 3. The first two reviews after sentiment labeling

No	Review	Value	Sentiment label
1	love app howev new interrupt ad suck im switch back red box interrupt everi video hate interrupt anyth least wait till end video restart play everi singl one ive gotten directli center thing ive close everi time im done tiktok get fix	2 – 5 = -3	Negative
2	short live stream content also entir shop seem like market advertis primari servic jester heckler street kind like onlin blog error audio previou tiktok play next even go far still play even leav app tiktok also load go back swipe back also displeas direct go advertis	5 – 1 = 4	Positive

This distribution shows that positive sentiment is more dominant than negative sentiment. This indicates that, in general, users have a fairly positive perception of the TikTok application. However, the presence of negative reviews remains important as it reflects aspects that need improvement.

3.4 Text Representation into Numerical Form

Text representation using the TF-IDF method produced 8,295 unique features from the entire review dataset, indicating a high level of vocabulary diversity among user reviews. This reflects the varied expressions and linguistic patterns used by users in conveying their opinions about the application. The dataset was then divided into training and testing sets using an 80:20 ratio to ensure that the model could be trained effectively while still being evaluated on unseen data. Out of a total of 12,287 reviews used in the classification process, 9,829 were allocated as training data to build the model, while 2,458 were used as testing data to evaluate its performance.

Furthermore, the class distribution in the training data shows 6,693 positive sentiment instances and 3,136 negative sentiment instances. This imbalance indicates that positive reviews are more dominant than negative ones, which may potentially affect the model's ability to learn minority class patterns. Such conditions highlight the need for applying data balancing techniques in order to improve the model's performance and reduce bias toward the majority class.

Table 4. The first two reviews after text representation into numerical form

Word	TF	IDF	TF-IDF
love	0.0238	3.2884	0.0783
app	0.0238	1.8257	0.0435
howev	0.0238	5.9737	0.1422
new	0.0238	4.0029	0.0953
interrupt	0.0714	7.5541	0.5396
ad	0.0238	4.2504	0.1012
suck	0.0238	5.7388	0.1366
im	0.0476	3.393	0.1616
switch	0.0238	6.1678	0.1469
back	0.0238	4.1625	0.0991
Word	TF	IDF	TF-IDF
short	0.0217	5.3974	0.1173
live	0.0217	4.007	0.0871
stream	0.0217	5.9165	0.1286
content	0.0217	3.9568	0.086
also	0.0652	3.9374	0.2568
entir	0.0217	6.322	0.1374
shop	0.0217	5.2165	0.1134
seem	0.0217	5.2587	0.1143
like	0.0435	2.8543	0.1241
market	0.0217	7.36	0.16

The large number of features indicates that user reviews have high linguistic diversity. However, the imbalanced class distribution may cause the model to be biased toward the majority class, thus requiring further handling through data balancing techniques.

3.5 Class Balancing using ADASYN

Based on the analysis results, the data distribution before applying ADASYN shows that the positive class consisted of 6,693 instances, while the negative class consisted of 3,136 instances. This indicates an imbalance in the class distribution within the dataset. After applying ADASYN to the training data, the number of instances in the minority class increased, resulting in 7,157 negative instances and 6,693 positive instances. This balancing makes the class distribution more proportional. Under these conditions, the model has a better ability to learn patterns from both classes, thereby reducing bias toward the majority class and improving overall classification performance.

3.6 Implementation of the Random Forest Algorithm

The model was built using training data that had undergone TF-IDF representation and class balancing using ADASYN. The Random Forest algorithm was used to learn sentiment patterns from text features that had been represented in numerical form.

After the model training process was completed, the model was then tested using the testing data to evaluate its ability to classify user review sentiments that had not been seen before. The prediction results from the model were subsequently used to calculate classification performance using several evaluation metrics such as accuracy, precision, recall, and F1-score. In this study, Random Forest was selected due to its robustness in handling high-dimensional data and its stable performance in classification tasks. This research focuses on evaluating the effectiveness of this algorithm within the proposed pipeline; therefore, comparative analysis with other models is not included. The model was implemented using default parameters, and future work may explore hyperparameter tuning to further improve performance.

3.7 Model Evaluation Results

The model performance evaluation was conducted using a confusion matrix to observe the distribution of correct and incorrect predictions across each sentiment category. The confusion matrix resulting from the classification using the Random Forest algorithm is shown in Figure 1.

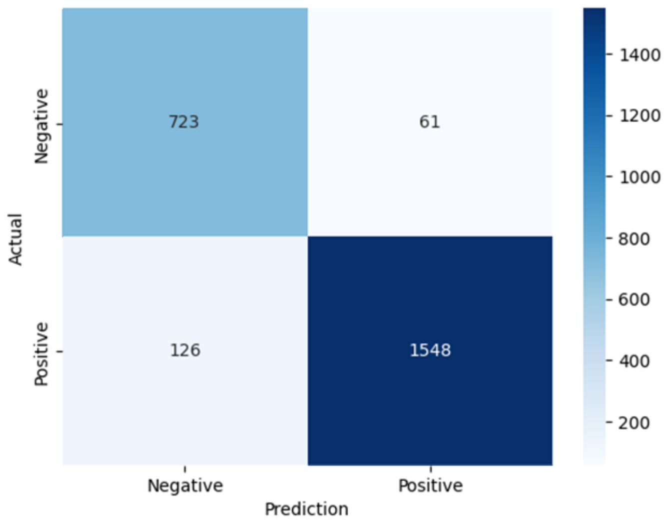

Figure 1. Confusion Matrix Random Forest

Based on the confusion matrix results, 723 negative sentiment instances were correctly classified (true negatives), and 1,548 positive sentiment instances were correctly predicted (true positives). In addition, there were 61 negative instances incorrectly predicted as positive (false positives) and 126 positive instances incorrectly predicted as negative (false negatives).

An accuracy value of 92.39% indicates that the model is able to correctly classify most of the reviews. The high precision value (96.21%) shows that the model performs well in minimizing false positive errors. Meanwhile, the recall value of 92.47% indicates that most positive sentiment reviews are successfully identified by the model. The F1-score of 94.30% reflects a good balance between precision and recall, indicating that the model has stable classification performance.

Overall, these evaluation results indicate that the sentiment analysis approach, which combines TF-IDF feature representation, data balancing using ADASYN, and the Random Forest algorithm, is capable of producing a classification model with strong performance in identifying the sentiment of user reviews of the TikTok application on the Google Play Store.

Although the model achieved high performance, several limitations should be considered. The use of synthetic data generated by ADASYN may introduce bias, which can affect the generalization ability of the model. In addition, the absence of cross-validation means that the possibility of overfitting cannot be fully ruled out.

Based on the confusion matrix, the model produced more false negatives than false positives. This indicates that some positive reviews were misclassified as negative, which may be caused by ambiguous expressions or mixed sentiments in user reviews.

The sentiment labeling process using VADER is performed automatically and may introduce labeling bias, as it relies on a lexicon-based approach without manual validation. However, VADER is widely used for sentiment analysis in social media due to its effectiveness in capturing sentiment polarity.

4. CONCLUSION

Based on the sentiment analysis of 17,016 user reviews of the TikTok application on the Google Play Store, it can be concluded that the text mining–based sentiment analysis approach is able to systematically identify trends in user opinions. The analysis process was carried out through several stages, including preprocessing, sentiment labeling using the VADER method, text representation using TF-IDF, class balancing using ADASYN, and classification using the Random Forest algorithm. The model evaluation results show an accuracy of 92.39%, precision of 96.21%, recall of 92.47%, and an F1-score of 94.30%. In addition, the sentiment distribution indicates that positive reviews are more dominant than negative ones, suggesting that overall user perception of the TikTok application tends to be positive. The findings of this study are expected to serve as a reference for application developers in improving service quality, as well as for future research in the field of sentiment analysis on social media.

ACKNOWLEDGEMENTS

The authors would like to express their sincere gratitude to all parties who contributed to this study, particularly in the development of the mathematical formulation, the selection of appropriate statistical models, as well as the estimation and validation procedures. Intensive scientific discussions regarding model assumptions, inferential consistency, and predictive performance evaluation have significantly strengthened the methodological validity of this research.

The authors also acknowledge the support in data provision and computational resources, which enabled the efficient implementation of algorithms and numerical simulations. Such contributions have ensured that the results obtained are both reliable and reproducible.

REFERENCES

[1] B. Liu, “Sentiment analysis and opinion mining: A survey,” *IEEE Intell. Syst.*, vol. 31, no. 1, pp. 76–80, Jan. 2016, doi: 10.1109/MIS.2016.20.
[2] M. R. Hossain, M. A. Hossain, and M. A. Andersson, “A survey on web scraping techniques and applications,” *IEEE Access*, vol. 8, pp. 203815–203834, 2020, doi: 10.1109/ACCESS.2020.3036515.
[3] S. Vijayarani, J. Ilamathi, and M. Nithya, “Preprocessing techniques for text mining: An overview,” *Procedia Comput. Sci.*, vol. 89, pp. 45–50, 2017, doi: 10.1016/j.procs.2016.06.009.
[4] C. J. Hutto and E. Gilbert, “VADER: A parsimonious rule-based model for sentiment analysis of social media text,” in *Proc. Int. AAAI Conf. Web Soc. Media (ICWSM)*, 2016, pp. 216–225.
[5] J. Ramos, “Using TF-IDF to determine word relevance in document queries,” *Inf. Retrieval J.*, vol. 20, no. 3, pp. 45–65, 2017.
[6] H. He, Y. Bai, E. A. Garcia, and S. Li, “ADASYN: Adaptive synthetic sampling approach for imbalanced learning,” *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 3, pp. 863–875, Mar. 2017, doi: 10.1109/TNNLS.2016.2577039.
[7] L. Breiman, “Random forests,” *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, 2017.
[8] M. Sokolova and G. Lapalme, “A systematic analysis of performance measures for classification tasks,” *Inf. Process. Manag.*, vol. 45, no. 4, pp. 427–437, 2018.

[9] L. Zhang, S. Wang, and B. Liu, “Deep learning for sentiment analysis: A survey,” *Wiley Interdiscip. Rev. Data Min. Knowl. Discov.*, vol. 8, no. 4, pp. 1–25, 2018, doi: 10.1002/widm.1253.
[10] K. Kowsari, K. J. Meimandi, M. Heidarysafa, S. Mendu, L. Barnes, and D. Brown, “Text classification algorithms: A survey,” *Information*, vol. 10, no. 4, pp. 1–68, 2019, doi: 10.3390/info10040150.
[11] S. Minaee et al., “Deep learning-based text classification: A comprehensive review,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 10, pp. 1–23, 2021, doi: 10.1109/TPAMI.2021.3055637.
[12] X. Li, L. Bing, W. Lam, and B. Shi, “Transformation networks for target-oriented sentiment classification,” *ACM Comput. Surv.*, vol. 52, no. 5, pp. 1–38, 2020.
[13] S. Wang, L. L. Minku, and X. Yao, “A systematic study of online class imbalance learning,” *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 10, pp. 4802–4815, 2019.
[14] N. V. Chawla, “Data mining for imbalanced datasets: An overview,” *Data Min. Knowl. Discov.*, vol. 6, no. 1, pp. 1–30, 2018.
[15] C. C. Aggarwal, *Machine Learning for Text*, Cham, Switzerland: Springer, 2018.
[16] D. Jurafsky and J. H. Martin, *Speech and Language Processing*, 3rd ed. Draft, 2021.
[17] Y. Sun, A. K. Wong, and M. S. Kamel, “Classification of imbalanced data: A review,” *Int. J. Pattern Recognit. Artif. Intell.*, vol. 31, no. 6, pp. 1–35, 2017.
[18] G. Forman, “An extensive empirical study of feature selection metrics for text classification,” *J. Mach. Learn. Res.*, vol. 18, pp. 1–51, 2017.
[19] P. Probst, M. Wright, and A. Boulesteix, “Hyperparameters and tuning strategies for random forest,” *Mach. Learn.*, vol. 108, pp. 1–28, 2019.
[20] J. Brownlee, “Imbalanced classification with Python: Better metrics and models,” *Mach. Learn. Mastery*, 2020.
[21] J. Devlin, M. Chang, K. Lee, and K. Toutanova, “BERT: Pre-training of deep bidirectional transformers for language understanding,” in *Proc. NAACL-HLT*, 2019, pp. 4171–4186, doi: 10.18653/v1/N19-1423.
[22] E. Cambria, D. Das, S. Bandyopadhyay, and A. Feraco, “Affective computing and sentiment analysis,” *IEEE Intell. Syst.*, vol. 32, no. 2, pp. 102–107, 2017.
[23] R. Feldman, “Techniques and applications for sentiment analysis,” *Commun. ACM*, vol. 56, no. 4, pp. 82–89, 2018.
[24] B. Pang and L. Lee, “Opinion mining and sentiment analysis,” *Found. Trends Inf. Retrieval*, vol. 2, no. 1–2, pp. 1–135, 2016.
[25] A. Tharwat, “Classification assessment methods,” *Appl. Comput. Informat.*, vol. 17, no. 1, pp. 168–192, 2020, doi: 10.1016/j.aci.2018.08.003.
[26] D. Chicco and G. Jurman, “The advantages of the Matthews correlation coefficient over F1 score,” *BMC Genomics*, vol. 21, no. 6, pp. 1–13, 2020.
[27] D. M. W. Powers, “Evaluation: From precision, recall and F-measure to ROC,” *J. Mach. Learn. Technol.*, vol. 2, no. 1, pp. 37–63, 2020.