Building of Informatics, Technology and Science (BITS)
Volume 6, No 4, March 2025 Page: 2240−2248
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v6i4.6677

Comparison of Random Forest and Decision Tree Methods for Emotion Classification based on Social Media Posts

Muhammad Abiyyu Tsaqif[1,*], Warih Maharani[2]

[1] Informatics, Informatics, Telkom University, Bandung, Indonesia
[2] Informatics, Data Science, Telkom University, Bandung, Indonesia
Email: [1,*]abiyyutsaqif@student.telkomuniversity.ac.id, [2]wmaharani@telkomuniversity.ac.id
Correspondence Author Email: abiyyutsaqif@student.telkomuniversity.ac.id
Submitted: **10/01/2025**; Accepted: **26/02/2025**; Published**: 01/03/2025**

Abstract−Social media platforms like X (formerly Twitter) have become essential for expressing emotions and opinions, making emotion classification a critical task with applications in mental health, public sentiment monitoring, and customer feedback analysis. This study compares Random Forest and Decision Tree algorithms for classifying emotions such as joy, sadness, anger, and fear which are from social media posts. Data collection involved crawling tweets and manual labeling. Preprocessing included tokenization, stemming, and stopword removal, with feature extraction using TF-IDF and Bag of Words. Experimental scenarios tested data split ratios, resampling for class balance, and parameter tuning. Decision Tree parameters included criterion (gini, entropy), max depth (none, fixed values), min samples split (2, 5), and min samples leaf (1, 2). Random Forest parameters tuned were n_estimators (100–400), max depth (none, fixed values), min samples split (2, 5, 10), and min samples leaf (1, 2). Results showed Random Forest achieving a maximum accuracy of 76.17%, outperforming Decision Tree's 72.62%. The combination of TF-IDF and Bag of Words delivered the highest accuracy for both models. This study underscores the importance of preprocessing, balanced datasets, and parameter optimization for effective emotion classification. The findings offer insights into advancing sentiment analysis and natural language processing, enabling practical applications in public sentiment tracking, customer experience enhancement, and crisis management.

Keywords: Bag of Words; Decision Tree; Emotion Classification; Random Forest; TF-IDF

1. INTRODUCTION

The rapid growth of the internet has enabled social media users to express their views, feelings, and thoughts, which can be analyzed through sentiment analysis, a computational study of emotions, judgments, and opinions [1]. Prinz identifies four basic human emotions: joy, sadness, fear, and anger [2]. According to Datareportal, with over 18 million active Indonesian users on X (Twitter) in 2022, the platform provides a space for emotional expression through text. Social media has become essential for expressing emotions and opinions, making emotion classification a critical task with applications in mental health support, public sentiment monitoring, and customer feedback analysis. However, indirect communication on social media often includes informal language, slang, and contextually ambiguous expressions, making it more challenging to identify emotions compared to face-to-face interactions. Addressing these challenges is vital for improving sentiment analysis systems, enabling their use in real-world applications such as detecting public distress during crises, tailoring marketing strategies based on customer emotions, and enhancing tools for mental health interventions. This highlights the need for robust classification systems capable of accurately analyzing emotions from social media data [3].

The focus of this research is about the emotion classification using the classification algorithms. There are some of classification algorithms that are commonly used, such as Decision Tree and Random Forest [4]. Decision Tree is an effective algorithm for doing classification and prediction. This algorithm has the ability to describe the conditions of different facts in the form of a decision tree structure [5]. A decision tree is a structure used to break down the large amounts of data to convert it into smaller groups of data [6]. This provides an understanding of which factors that influence the classification results, this making it suitable for emotion classification. On the other hand, the Random Forest algorithm was chosen as its ability to randomly combine a number of decision trees, which are then selected to determine the classification result [7]. Using this two algorithms, it's expected that more accurate emotion classification results can be obtained.

Several previous studies have demonstrated the effectiveness of Random Forest and Decision Tree models in classification tasks, but they often focus on limited scenarios, leaving gaps in understanding their performance under varying conditions. For instance, Setiawan et al. [8] utilized the Random Forest algorithm for sentiment analysis on GoFood review data, achieving an accuracy of 98.6% by focusing on binary sentiment polarity classification within a specific domain. This narrower focus simplifies the task compared to multi-class problems, where distinguishing between multiple categories introduces greater complexity. Similarly, Keskar et al. [9], applied a Decision Tree classifier for fake news detection on Twitter, achieving an accuracy of 70% by employing unigrams and TF-IDF for feature extraction. However, their study did not explore the impact of combining multiple feature extraction techniques, addressing data imbalance, or varying parameter settings.

These gaps underscore the need for further research to evaluate how these models perform under diverse testing scenarios. Specifically, there is a lack of studies examining the effects of different data split ratios, feature extraction methods such as TF-IDF, Bag of Words, Word2Vec, and their combinations, resampling techniques to handle imbalanced datasets, and parameter optimization. Exploring these factors is essential for more complex tasks like

Building of Informatics, Technology and Science (BITS)
Volume 6, No 4, March 2025 Page: 2240−2248
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v6i4.6677

multi-class emotion classification in noisy, informal social media data, where these elements could significantly influence model accuracy and robustness. By addressing these gaps, this study seeks to provide a more comprehensive understanding of the factors that affect the performance of Random Forest and Decision Tree models in such challenging contexts.

By addressing this gap, the main objective of this research is to implement the Random Forest and Decision Tree models for emotion classification from uploaded tweets to X social media, and to compare the performance of both algorithms in terms of efficiency and accuracy. In addition, this research is also focuses on evaluating the factors that influence classifying emotions based on users' uploading patterns on social media X. Therefore, this research will not only examine the application of the models, but also comparing their performance and the classification of the upload patterns into emotion classification.

2. RESEARCH METHODOLOGY

The research process for emotion classification based on posts on social media X starts with data crawling which then forms a dataset. The dataset goes through a data labeling process to provide appropriate annotations. After that, the data is processed through preprocessing steps, including case folding, cleansing, tokenization, normalization, stemming, and stopword removal, to improve the quality of the data for further analysis. The preprocessed dataset is then splitted into training data (train data) and test data (test data). Text representation using methods such as TF-IDF, Bag of Words, and Word2Vec are applied to both subsets of data. The training data is used to build models with algorithms such as Random Forest and Decision Tree. Once the model is built, evaluation is done using the test data to measure the performance of the model. The research process can be seen in Figure 1.

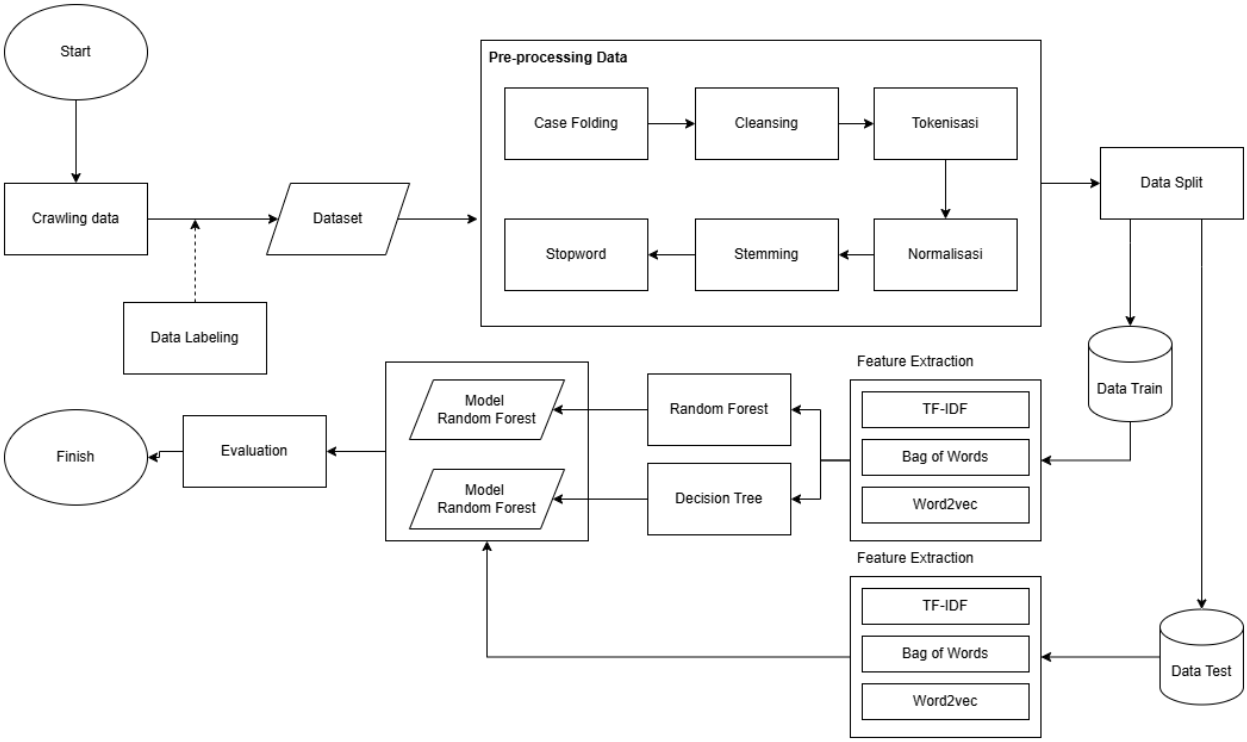

Figure 1. Research Process

2.1 Data Collection

In machine learning, data collection is an important process that involves acquiring and organizing data sets to train models efficiently [10]. In this research, the data is taken from user tweets on social media X by using several stages in data collection. First, data was obtained using a Microsoft Form questionnaire to collect X social media usernames from users who were willing to share their tweets as a data set for research. After that, the data set is collected by crawling tweets from user accounts that have been collected through the questionnaire, with the condition that the account is public.

2.2 Data Labelling

The labeling of this data was done manually by six people, as emotions are inherently subjective, so manual labeling was the most appropriate approach [11]. The labeled emotions were categorized into four classes such as joy, anger, sad, and fear.

Building of Informatics, Technology and Science (BITS)
Volume 6, No 4, March 2025 Page: 2240−2248
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v6i4.6677

2.3 Preprocessing Data

Data preprocessing transforms raw data into a suitable format for analysis, enhancing model performance and accuracy [12]. Steps like cleaning, normalization, and feature selection improve data quality, ensuring reliability for emotion classification models [13]. Key preprocessing steps are summarized in Table 1.

Table 1. Prepocessing Steps

Steps	Result
Full Text	Provides raw crawled text that has not been preprocessed.
Case Folding	Convert all characters in the text to lowercase for consistency.
Cleansing	Remove irrelevant characters such as punctuation marks, numbers, or symbols.
Tokenization	Break the text into word units or tokens to facilitate further analysis.
Normalization	Mengubah kata tidak baku atau slang menjadi kata standar sesuai dengan kamus bahasa.
Stemming	Mengembalikan kata ke bentuk dasar atau kata dasarnya untuk menyederhanakan representasi.
Stopword	Menghapus kata-kata umum yang tidak memiliki makna signifikan dalam konteks analisis teks.

2.4 TF-IDF

Term Frequency - Inverse Document Frequency (TF-IDF) is a method of weighting words to determine the importance of a word in a document or set of documents [14]. Term Frequency (TF) measures a word's frequency in a document, while Inverse Document Frequency (IDF) evaluates its importance across documents. TF-IDF has applied in various contexts, such as document classification and sentiment analysis, demonstrating its effectiveness in these domains [15]. The combined TF-IDF scheme is widely used in text mining and information retrieval to determine word significance . TF and IDF are calculated using equations (1) and (2), where N represents the total number of documents in the corpus, and DF_t denotes the document frequency of term t which is the number of documents containing term t.

$$IDF_t = \log\left(\frac{N}{DF_t}\right) \tag{1}$$

$$\omega_t = TF_t \cdot IDF_t \tag{2}$$

2.5 Bag of Words

Bag of Words (BoW) is a text processing technique that represents documents as unordered word collections, focusing on word frequency while ignoring grammar and word order, widely used in natural language processing tasks like text classification [16][17]. The advantage of BoW is its ability to convert text into a numerical format that can be processed by machine learning algorithms [18]. Bag of Words transforms text into a vector based on the frequency of occurrence of words in a document [19]. BoW calculation is based on formula (3).

$$\boldsymbol{vi} = [f(\omega 1, di), f(\omega 2, di), \ldots, f(\omega m, di)] \tag{3}$$

2.6 Word2Vec

Word2Vec is a word embedding model that transforms words into high-dimensional vectors to capture semantic relationships between them. According to Lierler and Schüller [20], Word2Vec represents each word x as a vector $(vec(x))$ in a high-dimensional space, enabling mathematical operations like vector addition and subtraction to identify semantic patterns. For example, in analogy tasks such as determining that “man is to woman as king is to queen,” the model finds the word $\hat{x}$ that maximizes similarity in the vector space. Cosine similarity is used to measure the semantic relationship between words based on their vector representations, as demonstrated in the equations (4) and (5).

$$\hat{x} = \text{ARGMAX}_{x' \in V} \text{sim}\big(x', \text{vec(king)} + \text{vec(woman)} - \text{vec(man)}\big) \tag{4}$$

$$\text{sim}(a, b) = \frac{\text{vec}(a) \cdot \text{vec}(b)}{|\text{vec}(a)||\text{vec}(b)|} \tag{5}$$

2.7 Random Forest

Random Forest is a classification method comprising a collection of decision trees, where each tree is built using independent and identically distributed random vectors. The algorithm determines the dominant class of an input by allowing each tree to vote [21]. Figure 2 illustrates how the Random Forest algorithm works.

Building of Informatics, Technology and Science (BITS)
Volume 6, No 4, March 2025 Page: 2240−2248
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v6i4.6677

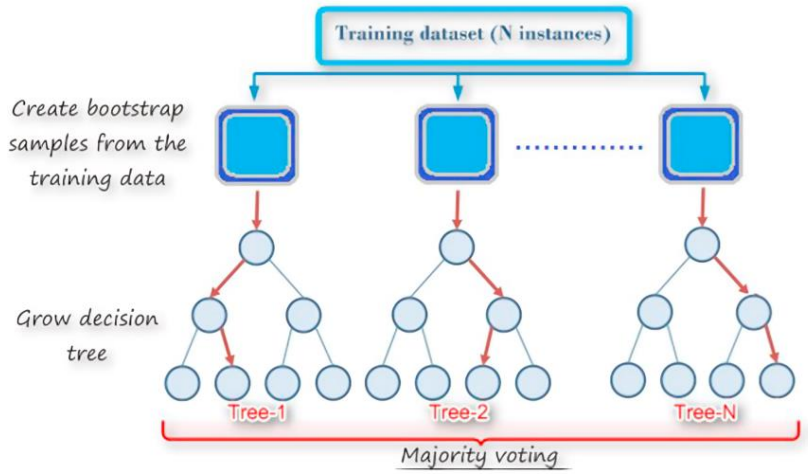

Figure 2. Random Forest [22]

Random Forest enhances prediction accuracy by combining multiple decision trees, with each tree classifying inputs independently and results aggregated. It is highly effective for large datasets and offers robust performance [22]. The Random Forest algorithm employs bootstrap sampling to create prediction trees. In this process, each tree uses randomly selected predictors and combines their majority classification or regression results [23].

2.7 Decision Tree

A Decision Tree is a classification algorithm that divides data iteratively, with a structure comprising a root node (no incoming branches), internal nodes (with outgoing branches), and leaf nodes (no outgoing branches), where each internal node splits data based on attribute values [24]. An example of a Decision Tree is shown in Figure 3.

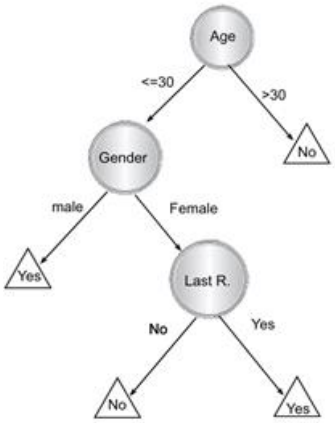

Figure 3. Decison Tree [24]

In Figure 3, "Age" is the root node, "Gender" and "Last R." are internal nodes, and "yes" or "no" leaves show classification results. The Gini Index, a commonly used impurity measure in decision trees, evaluates the quality of splits by prioritizing attributes that yield purer subsets, with lower values signifying better splits and improved classification performance [25]. To assess feature importance, the algorithm uses the Gini Index, which evaluates the effectiveness of a feature in classification. The Gini Index of a node n is computed using formula (6) [26].

$$Gini(D) = 1 - \sum_{i=1}^{m} p_i^2 \quad (6)$$

2.7 Evaluation

The evaluation measures the effectiveness of the developed model using metrics such as precision, recall, F1-score, and accuracy [27]. This study compares the performance of Random Forest and Decision Tree algorithms, focusing on analyzing their accuracy, precision, recall, and F1-score.

$$Precision = \frac{\text{TP}}{\text{TP+FP}} \quad (7)$$

$$Recall = \frac{\text{TP}}{\text{TP+FN}} = \frac{\text{TN}}{\text{P}} \quad (8)$$

Building of Informatics, Technology and Science (BITS)
Volume 6, No 4, March 2025 Page: 2240−2248
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v6i4.6677

$$F1 - score = \frac{2\times(Recall\times Precision)}{Recall+Precision} \quad (9)$$

$$Accuracy = \frac{\text{TP+TN}}{\text{TP+FP+TN+FN}} \quad (10)$$

The performance of a classification model is evaluated using several metrics. Precision measures the proportion of correct positive predictions out of all positive predictions, reflecting the relevance of the results as shown in equation (7). Recall compares the correct positive predictions with the total actual positive cases to measure the detection rate of positive cases, as shown in equation (8). The F1-score combines precision and recall into a single metric, balancing relevance and sensitivity, as shown in equation (9). Finally, Accuracy quantifies the proportion of correct predictions (both positive and negative) out of all predictions made by the model, as shown in equation (10). These metrics collectively provide a comprehensive evaluation of the model's performance.

3. RESULT AND DISCUSSION

3.1 Data Collection and Data Labelling

The data in this study was obtained through a crawling process on platform X, resulting in 8,978 tweets as the main corpus. Furthermore, the raw data was manually labeled into four emotion categories, namely joy, sadness, anger, and fear, based on the emotional context contained in each tweet. The labeling process is done to ensure accuracy in data grouping so that it can be used as the basis for training text-based emotion classification models. The results of the distribution of emotion labels on the data are shown in Table 2.

Table 2. Data Distribution

Joy	Sad	Anger	Fear
2,723	2,001	2,492	1,762

3.2 Data Preprocessing

The next step in this research is to preprocess the data to improve the quality of the text before it is used in model analysis and training. Preprocessing is done through several stages summarized in Table 3, including case folding, cleansing, tokenization, normalization, stemming, and stopword removal. These stages aim to remove irrelevant elements, such as punctuation and meaningless words, as well as aligning word forms to be more consistent with the needs of the analysis.

Table 3. Prepocessing Stages

Steps	Result
Full Text	"Kadang gue suka mikir, kita ini nikmatin masa muda, atau ngehancurin masa depan?"
Case Folding	"kadang gue suka mikir, kita ini nikmatin masa muda, atau ngehancurin masa depan?"
Cleansing	kadang gue suka mikir kita ini nikmatin masa muda atau ngehancurin masa depan
Tokenization	["kadang", "gue", "suka", "mikir", "kita", "ini", "nikmatin", "masa", "muda", "atau", "ngehancurin", "masa", "depan"]
Normalization	["kadang", "saya", "suka", "mikir", "kita", "ini", "menikmati", "masa", "muda", "atau", "menghancurkan", "masa", "depan"]
Stemming	["kadang", "saya", "suka", "mikir", "kita", "ini", "nikmat", "masa", "muda", "atau", "hancur", "masa", "depan"]
Stopword	["kadang", "saya", "suka", "mikir", "nikmat", "muda", "hancur"]

3.3 Scenario and Test Results

After preprocessing data, the next step is to perform various test scenarios. There are four test scenarios, including:

a. Testing several split ratios of test data and training data.
b. Testing several feature extraction methods such as TF- IDF, Bag of Words (BoW), and Word2Vec, as well as their combinations.
c. Testing resampling methods.
d. Testing several parameter tuning options for Random Forest and Decision Tree.

3.1.1 First test scenario

The First test scenario is using several split ratios to see the effect of the division ratio on the Random Forest and Decision Tree models. In its application, this test does not use feature extraction. Therefore, a word_encoder is used so that the data is numeric so that it can be evaluated using both of the machine learning models. The results of the data split ratio tests can be seen in Table 4.

Building of Informatics, Technology and Science (BITS)
Volume 6, No 4, March 2025 Page: 2240−2248
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v6i4.6677

Table 4. Split Ratio Tests

Split Ratio	Model	Accuracy (%)
90:10	**Random Forest**	**36.86**
90:10	**Decision Tree**	**35.75**
80:20	Random Forest	34.02
80:20	Decision Tree	34.02
70:30	Random Forest	33.33
70:30	Decision Tree	33.04
60:40	Random Forest	34.24
60:40	Decision Tree	33.66
50:50	Random Forest	33.79
50:50	Decision Tree	33.59

The accuracy results for the Random Forest and Decision Tree models show the best performance at a 90:10 split ratio, with Random Forest achieving 36.86% and Decision Tree 35.75%.

3.1.2 Second test scenario

The second test scenario uses several combinations of feature extraction methods to evaluate their accuracy performance. The methods used include TFIDF, BoW (Bag of Words), Word2Vec, as well as a combination of these methods. Each method is tested with two different models, Random Forest and Decision Tree. The Table 5 shows the accuracy results obtained for each combination of method and model.

Table 5. Feature Extraction Tests

Feature Extraction	Model	Accuracy (%)
TF-IDF	Random Forest	68.15
TF-IDF	Decision Tree	64.69
BoW	Random Forest	68.70
BoW	**Decision Tree**	**66.70**
Word2Vec	Random Forest	46.10
Word2Vec	Decision Tree	37.63
TF-IDF + BoW	**Random Forest**	**69.48**
TF-IDF + BoW	Decision Tree	64.92
TF-IDF + Word2Vec	Random Forest	57.34
TF-IDF + Word2Vec	Decision Tree	57.57
BoW + Word2Vec	Random Forest	57.90
BoW + Word2Vec	Decision Tree	60.35
TF-IDF + BoW + Word2Vec	Random Forest	61.35
TF-IDF + BoW + Word2Vec	Decision Tree	59.02

From the results obtained, the combination of TF-IDF + BoW with Random Forest produced the highest accuracy of 69.48%. For Decision Tree, the BoW combination yielded the highest accuracy of 66.70%, indicating that although no single method dominates, certain combinations perform better on different models.

3.1.3 Third test scenario

The third scenario test is to use resampling. Resampling is used with the aim that the distribution of data between emotion labels is evenly distributed. In the data distribution, the smallest data is in the ’fear’ emotion label, which is at 1,762. While the largest data is on the ’joy’ emotion label, which is at 2,723. Therefore, in this case the resampling used is undersampling with the aim of leveling the data of other emotion labels with the ’fear’ emotion label. Thus, all emotion labels were changed to 1,762. The result after doing the resampling can be seen on Table 6.

Table 6. Data Distribution After Resampling

Joy	Sad	Anger	Fear
1,762	1,762	1,762	1,762

After resampling, test scenarios were conducted using both models to see the effect of resampling. The results of this test can be seen in the Table 7.

Table 7. Resampling Tests

Feature Extraction	Model	Accuracy (%)
Before Resampling		
BoW	Decision Tree	64.69
TF-IDF + BoW	Random Forest	68.70

Building of Informatics, Technology and Science (BITS)
Volume 6, No 4, March 2025 Page: 2240−2248
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v6i4.6677

After Resampling		
BoW	Decision Tree	64.69
TF-IDF + BoW	Random Forest	68.70

There was an increase in accuracy for both models. The Decision Tree model has an increase in accuracy from 66.70% to 72.62%. Then, for the Random Forest model, the accuracy increased from 69.40% to 75.32%.

3.1.4 Fourth test scenario

The fourth test scenario is to use parameter tuning for both models. For the Decision Tree model, there are several parameters tested, such as, criterion, max depth, min samples split, and min samples leaf. As for the Random Forest model, several parameters were tested such as, n estimator, max depth, min samples split, and min samples leaf. The results of the test of several parameters for the Decision Tree model and the Random Forest model can be seen in Table 8 and Table 9.

Table 8. Parameter Tuning for Decion Tree

Step	Parameter tested	Accuracy (%)
1	**criterion: gini (selected)**	**72.62**
	criterion: entropy	**70.21**
2	**max depth: none (selected)**	**72.62**
	max depth: 10	**47.23**
3	**min samples split: 2 (selected)**	**72.62**
	min samples split: 5	**71.77**
4	**min samples leaf: 1 (selected)**	**72.62**
	min samples leaf: 2	**69.78**

The Table 8 shows the parameter tuning results for the Decision Tree Classifier, which aims to improve the accuracy of the model. In the first step, the criterion parameter was tested with two options, such as gini and entropy. The results show that gini gives the highest accuracy of 72.62%, so it is chosen as the best parameter. Furthermore, in the max depth parameter, the none option resulted in an accuracy of 72.62%, better than the maximum depth of 10 which only reached 47.23%. In the next step, the min samples split parameter was tested with values of 2 and 5, where the value of 2 gave a higher accuracy of 72.62%. Finally, the min samples leaf parameter was tested with values of 1 and 2, where the value of 1 yielded the highest accuracy of 72.62%. The best parameter combination from this tuning is criterion = gini, max depth = none, min samples split = 2, and min samples leaf = 1, with a maximum accuracy achieved of 72.62%.

Table 9. Parameter Tuning for Random Forest

Step	Parameter tested	Accuracy (%)
1	n estimators: 100	**75.32**
	n estimators: 200	**75.74**
	n estimators: 300 (selected)	**75.89**
	n estimators: 400	**75.46**
2	**max depth: none (selected)**	**75.89**
	max depth: 10	**67.09**
3	min samples split: 2	**75.89**
	min samples split: 5 (selected)	**76.17**
	min samples split: 10	**75.18**
4	**min samples leaf: 1 (selected)**	**76.17**
	min samples leaf: 2	**73.76**

The Table 9 shows the parameter tuning results for the Random Forest Classifier to improve the accuracy of the model. In the first step, the parameter n estimators, which determines the number of trees in the forest, was tested with four values: 100, 200, 300, and 400. The best value was 300, with the highest accuracy of 75.89%. Furthermore, for the max depth parameter, two options were tested: none and a maximum depth of 10. The none option produced the highest accuracy of 75.89% and was chosen as the best. For the min samples split parameter, which specifies the minimum number of samples to split an internal node, values of 2, 5, and 10 were tested. The value of 5 gave the highest accuracy of 76.17%, so it was chosen as the optimal parameter. Finally, the min samples leaf parameter, which specifies the minimum number of samples on each leaf, was tested with values of 1 and 2. The value of 1 gave the highest accuracy of 76.17% and was chosen as the best. The optimal parameter combination resulting from this tuning was n estimators = 300, max depth = none, min samples split = 5, and min samples leaf = 1, with a maximum accuracy of 76.17%.

Building of Informatics, Technology and Science (BITS)
Volume 6, No 4, March 2025 Page: 2240−2248
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v6i4.6677

3.4 Discussion

This study investigated the performance of Random Forest and Decision Tree models in emotion classification, using various split ratios, feature extraction methods, resampling techniques, and parameter tuning strategies. In the first scenario, the 90:10 split ratio yielded the highest accuracy for both models, with Random Forest achieving 36.86% and Decision Tree reaching 35.75%. This finding suggests that a larger training set, while maintaining a small test set, facilitates better learning for these models. The second scenario tested different feature extraction techniques, such as TF-IDF, Bag of Words (BoW), and Word2Vec, along with their combinations. The combination of TF-IDF and BoW achieved the best accuracy for Random Forest 69.48%, while the BoW method alone performed best for Decision Tree 66.70%. This demonstrates the importance of feature engineering in improving model performance, as combinations of methods can capture more diverse textual characteristics. In the third scenario, undersampling was employed to balance the data distribution across emotion labels, equalizing the number of samples for each label. This step significantly improved model accuracy, with Decision Tree accuracy rising from 66.70% to 72.62%, and Random Forest accuracy increasing from 69.40% to 75.32%. This outcome highlights the impact of balanced datasets in reducing bias and improving model effectiveness. Lastly, parameter tuning was applied to optimize both models. For Decision Tree, the best parameter combination achieved an accuracy of 72.62%, while for Random Forest, fine-tuning resulted in an accuracy of 76.17%.

The results of this study align with findings from the literature but also present unique contributions. Keskar et al. [9] conducted a study using N-gram analysis and a Decision Tree classifier for fake news detection on Twitter, achieving an accuracy of 70%. While their study focused on unigrams and TF-IDF for feature weighting, this study explored various combinations of TF-IDF, BoW, and Word2Vec, achieving higher accuracy for Decision Tree (72.62%) through balanced data and parameter tuning. This difference illustrates the potential of combining feature extraction methods to enhance model performance. In another study, Setiawan et al. [8] utilized Random Forest for sentiment analysis, achieving an impressive accuracy of 98.6% by employing TF-IDF and sentiment polarity analysis on GoFood review data. Their focus on a specific domain (e-commerce) and a binary sentiment classification task contributed to their high accuracy. In contrast, this study tackled the more complex task of multi-class emotion classification, achieving a maximum accuracy of 76.17% for Random Forest. While the accuracy is comparatively lower, it reflects the added complexity of distinguishing multiple emotional categories.

4. CONCLUSION

This study evaluates the effectiveness of Random Forest and Decision Tree algorithms in classifying emotions from social media posts, leveraging various preprocessing and feature extraction techniques such as tokenization, stemming, TF-IDF, and Bag of Words. Random Forest demonstrated superior performance, achieving a maximum accuracy of 76.17% after parameter tuning, while Decision Tree reached an accuracy of 72.62% under optimal conditions. Parameter tuning played a critical role in enhancing both models' performance, with key adjustments to parameters such as criterion, max depth, min samples split, and min samples leaf for Decision Tree, and n estimators, max depth, min samples split, and min samples leaf for Random Forest. The results emphasize the importance of preprocessing, balanced datasets, and parameter optimization. Both models showed significant improvements post-tuning, showcasing their capability to capture emotional nuances effectively. These findings highlight the potential of combining advanced feature extraction and tuning techniques for emotion classification.

REFERENCES

[1] B. Liu, “Sentiment Analysis and Opinion Mining” Morgan & Claypool Publisher, 2012
[2] J. Prinz, “Which Emotions Are Basic?,” Oxford University Press, 2004. doi: 10.1093/acprof:oso/9780198528975.003.0004.
[3] A. Al Maruf, F. Khanam, M. M. Haque, Z. M. Jiyad, M. F. Mridha, and Z. Aung, “Challenges and Opportunities of Text-Based Emotion Detection: A Survey,” *IEEE Access*, vol. 12, pp. 18416–18450, 2024, doi: 10.1109/ACCESS.2024.3356357.
[4] T. Heričko and B. Šumak, “Commit Classification into Software Maintenance Activities: A Systematic Literature Review,” in *Proceedings - International Computer Software and Applications Conference*, 2023. doi: 10.1109/COMPSAC57700.2023.00254.
[5] F. Aaboub, H. Chamlal, and T. Ouaderhman, “Analysis of the prediction performance of decision tree-based algorithms,” *2023 International Conference on Decision Aid Sciences and Applications, DASA 2023*, pp. 7–11, 2023, doi: 10.1109/DASA59624.2023.10286809.
[6] D. Septhya *et al.*, “Implementasi Algoritma Decision Tree dan Support Vector Machine untuk Klasifikasi Penyakit Kanker Paru,” *MALCOM: Indonesian Journal of Machine Learning and Computer Science*, vol. 3, no. 1, pp. 15–19, May 2023, doi: 10.57152/MALCOM.V3I1.591.
[7] T. T. Huynh-Cam, L. S. Chen, and H. Le, “Using decision trees and random forest algorithms to predict and determine factors contributing to first-year university students’ learning performance,” *Algorithms*, vol. 14, no. 11, Nov. 2021, doi: 10.3390/a14110318.
[8] 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.

Building of Informatics, Technology and Science (BITS)
Volume 6, No 4, March 2025 Page: 2240−2248
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v6i4.6677

[9] D. Keskar, S. Palwe, and A. Gupta, "Fake News Classification on Twitter Using Flume, N-Gram Analysis, and Decision Tree Machine Learning Technique," Proceeding of International Conference on Computational Science and Applications, pp. 139–147, 2020, doi: 10.1007/978-981-15-0790-8_15.
[10] H. Taherdoost, "Data Collection Methods and Tools for Research; A Step-by-Step Guide to Choose Data Collection Technique for Academic and Business Research Projects," International Journal of Academic Research in Management (IJARM), 2021. [Online]. Available: https://hal.science/hal-03741847v1
[11] V. Vine, E. E. Bernstein, and S. Nolen-Hoeksema, "Less is more? Effects of exhaustive vs. minimal emotion labelling on emotion regulation strategy planning," *Cogn Emot*, vol. 33, no. 4, 2019, doi: 10.1080/02699931.2018.1486286.
[12] P. Li, Z. Chen, X. Chu, and K. Rong, "DiffPrep: Differentiable Data Preprocessing Pipeline Search for Learning over Tabular Data," *Proceedings of the ACM on Management of Data*, vol. 1, no. 2, pp. 1–26, Jun. 2023, doi: 10.1145/3589328.
[13] K. Goyle, Q. Xie, and V. Goyle, "DataAssist: A Machine Learning Approach to Data Cleaning and Preparation," Jul. 2023, [Online]. Available: http://arxiv.org/abs/2307.07119
[14] A. Jalilifard, V. F. Caridá, A. F. Mansano, R. S. Cristo, and F. P. C. da Fonseca, "Semantic Sensitive TF-IDF to Determine Word Relevance in Documents," Advances in Computing and Network Communications, vol. 735, Jan. 2020, doi: 10.1007/978-981-33-6977-1.
[15] W. N. Ibrahem Al-Obaydy, H. A. Hashim, Y. AbdulKhaleq Najm, and A. A. Jalal, "Document classification using term frequency-inverse document frequency and K-means clustering," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 27, no. 3, pp. 1517–1524, Sep. 2022, doi: 10.11591/ijeecs.v27.i3.pp1517-1524.
[16] D. Yan, K. Li, S. Gu, and L. Yang, "Network-Based Bag-of-Words Model for Text Classification," *IEEE Access*, vol. 8, pp. 82641–82652, 2020, doi: 10.1109/ACCESS.2020.2991074.
[17] U. K. Singh, B. Prabhu Shankar, R. Chinnaiyan, and N. Jain, "Machine Learning-Based Text Categorization with Bag of Words," *Lecture Notes in Electrical Engineering*, vol. 1194, pp. 577–587, 2024, doi: 10.1007/978-981-97-2839-8_40.
[18] W. A. Qader, M. M. Ameen, and B. I. Ahmed, "An Overview of Bag of Words;Importance, Implementation, Applications, and Challenges," *Proceedings of the 5th International Engineering Conference, IEC 2019*, pp. 200–204, Jun. 2019, doi: 10.1109/IEC47844.2019.8950616.
[19] D. Intan Af *et al.*, "Pengaruh Parameter Word2Vec terhadap Performa Deep Learning pada Klasifikasi Sentimen," vol. 6, no. 3, 2021, doi: https://doi.org/10.30591/jpit.v6i3.3016.
[20] J. Liao, Y. Huang, H. Wang, and M. Li, "Matching Ontologies with Word2Vec Model Based on Cosine Similarity," pp. 367–374, 2021, doi: 10.1007/978-3-030-76346-6_34.
[21] L. Breiman, "Random Forests," Machine Learning, 2001. doi: doi.org/10.1023/A:1010933404324.
[22] Y. Al Amrani, M. Lazaar, and K. E. El Kadirp, "Random forest and support vector machine based hybrid approach to sentiment analysis," in *Procedia Computer Science*, Elsevier B.V., 2018, pp. 511–520. doi: 10.1016/j.procs.2018.01.150.
[23] X. Chen, D. Yu, and X. Zhang, "Optimal Weighted Random Forests," May 2023, [Online]. Available: http://arxiv.org/abs/2305.10042
[24] L. Rokach and O. Maimon, "Decision Trees," Data Mining and Knowledge Discovery Handbook, 2005. doi: https://doi.org/10.1007/0-387-25465-X_9.
[25] G. Zhang and A. Gionis, "Regularized impurity reduction: Accurate decision trees with complexity guarantees," *Data Min Knowl Discov*, vol. 37, no. 1, pp. 434–475, Aug. 2022, doi: 10.1007/s10618-022-00884-7.
[26] L. Ceriani and P. Verme, "The origins of the Gini index: Extracts from Variabilità e Mutabilità (1912) by Corrado Gini," *J Econ Inequal*, vol. 10, no. 3, pp. 421–443, Sep. 2012, doi: 10.1007/s10888-011-9188-x.
[27] P. Singh, N. Singh, K. K. Singh, and A. Singh, "Diagnosing of disease using machine learning," *Machine Learning and the Internet of Medical Things in Healthcare*, pp. 89–111, Jan. 2021, doi: 10.1016/B978-0-12-821229-5.00003-3.