Published online on the journal's webpage: **http://journal.ittelkom-pwt.ac.id/index.php/dinda**

Journal of Dinda

Data Science, Information Technology, and Data Analytics

Vol. 5 No. 2 (2025) 230 - 240 E-ISSN: 2809-8064

Classification of Indonesian Disasters with Decision Trees Based on Spatial and Text Data

Zacky Innova[1*], Ridwan Ramadhan[2], Ragil Raditya[3], Ary Prabowo[4]

[1*,2,3,4]Teknik Informatika, Ilmu Komputer, Universitas Esa Unggul

[1*]zackyinova1@student.esaunggul.ac.id, [2]ramadhanridwan660@student.esaunggul.ac.id, [3]ragil.raditya2005@student.esaunggul.ac.id, [4]ary.prabowo@esaunggul.ac.id

Abstract

Indonesia is one of the countries with a very high vulnerability to natural disasters. Common types of disasters include earthquakes, floods, landslides, volcanic eruptions, and others. This condition is caused by Indonesia's geographical position, which lies between three major tectonic plates, as well as its tropical climate that is prone to extreme weather. Therefore, a system capable of automatically and accurately classifying types of disaster events is needed to support fast and precise decision-making in emergency response. This study aims to develop a multi-class classification model for disaster events based on location, time, and cause of occurrence. The 'cause' column, which contains textual data, is processed using the Term Frequency–Inverse Document Frequency (TF-IDF) technique to convert it into a numerical representation. Two machine learning algorithms, Decision Tree and Random Forest, are compared to evaluate classification performance on an open dataset from the Indonesian National Disaster Management Authority (BNPB), covering data from 2020 to 2024. The Decision Tree model achieved an accuracy of 87% with a macro F1-score of 0.60, but showed weaknesses in classifying disaster categories with minority data. As a comparison, the Random Forest algorithm was applied and showed improved performance with an accuracy of 91% and a more balanced and stable macro F1-score. These results indicate that Random Forest is more effective in handling class imbalance. Future research can integrate oversampling techniques, advanced ensemble methods, and spatial modeling to further improve accuracy. This study is expected to serve as a foundation for the development of more adaptive and efficient disaster classification systems.

Keywords: *Natural Disaster, Decision Tree, Machine Learning, Random Forest, TF-IDF*

© 2025 Journal of DINDA

1. Introduction

According to data from the Indonesian National Disaster Management Authority (BNPB), Indonesia is one of the countries most frequently affected by natural disasters. The high frequency of events such as floods, earthquakes, landslides, and volcanic eruptions highlights the need for a system that can quickly and accurately classify the types of disasters. Such a classification system is essential to support effective monitoring, timely decision-making, and to mitigate the impact on affected communities.

Disaster data typically includes information such as the time of occurrence, location, and cause most of which are recorded in textual form. However, manually classifying this type of data is time-consuming and prone to human error. Therefore, an automated approach using machine learning algorithms is required, as these are capable of processing both numerical and textual data.

In this study, the textual data in the "cause" column was converted into numerical form using the Term Frequency Inverse Document Frequency (TF-IDF) technique. To build the classification model, the Decision Tree algorithm was employed due to its interpretability and its ability to handle categorical features. However, to more comprehensively evaluate the effectiveness of this method, the study also compares it with the Random Forest algorithm, which is known for its superior performance in handling imbalanced data.

By comparing these two algorithms, this research aims to contribute to the development of a disaster classification system that is more accurate and adaptive

Received: 25-07-2025 | Accepted: 16-08-2025 | Published: 19-08-2025

to the unequal distribution of data across different disaster categories.

2. Research Methods

This study employs a quantitative approach using an experimental method to develop a natural disaster classification model using the Decision Tree algorithm based on spatial and textual data. The research methodology is systematically designed with clear stages, ranging from data collection to model evaluation, to ensure accurate results that can be reproduced by other researchers [1][2].

2.1. Literature Review

2.1.1. TF-IDF

TF-IDF (Term Frequency–Inverse Document Frequency) is a technique commonly used in text processing. The purpose of TF-IDF is to measure the importance of a word within a document relative to a collection or corpus [6][8][10].

$$TF-IDF(t,d) = TF\ (t,d) * IDF(t) \tag{1}$$

Description:

a. TF(t, d) (Term Frequency): How frequently the term *t* appears in document *d*
b. IDF(t) (Inverse Document Frequency): How rare the term *t* is across the entire collection of documents

1. TF (Term Frequency)

Term Frequency (TF) is a component used to calculate how often a word appears in a document. The more frequently a word appears, the higher its weight. However, to maintain balance, the frequency can be divided by the total number of words in the document, ensuring a fair value for longer documents [13].

$$TF(t,d) = \frac{Count\ of\ term\ t}{Total\ words\ in\ d} \tag{2}$$

2. IDF (Inverse document frequency)

Inverse Document Frequency (IDF) is a component commonly used to measure how unique a word is across documents. Words that are rare or appear in fewer documents tend to have higher IDF values [9][10]. IDF is calculated by dividing the total number of documents by the number of documents that contain the term.

$$DF(t) = \log(\frac{N}{df(t)+1}) \tag{3}$$

Description :

A. N : total number of documents in the dataset
B. $df(t)$: number of documents containing the term *t*
C. The addition of 1 in the denominator is to avoid division by zero

2.1.2. Decision Tree

The Decision Tree algorithm is a machine learning algorithm that can be used to make decisions using a tree-like structure [7][11][12]. This algorithm works by selecting features from the dataset based on the most significant attribute and uses methods such as Information Gain, Gini Index, or Gain Ratio to split the criteria.

1. Entrophy

$$Entrophy\ (S) = \sum_{i=1}^{n} -p(i)\ * \log_2 pi \tag{4}$$

Description :

a. S : Set of data
b. N : Number of classes in the dataset
c. $P(i)$: Probability of the *i*-th class in dataset *S*

2. Gain

$$Gain(S,A) = Entrophy(S) - \sum_{i=1}^{n} \frac{|S_i|}{|S|} * Entrophy(Si) \tag{5}$$

Decsription :

c. S : Initial dataset
d. A : Feature used for splitting
e. N : Number of data instances in the *i*-th partition
f. $|S_i|$: Number of instances in the *i*-th partition
g. $|s|$: Total number of instances in *S*

2.1.3. Evaluation Metrics in Classification

Evaluation metrics are parameters used to measure the performance of a model in a specific task [15]. One of the most fundamental evaluation methods is the Confusion Matrix, which is used to visualize classification results by comparing actual labels with the predicted labels generated by the model. The components of the Confusion Matrix include:

1. True Positive (TP) : Total number of correct positive predictions
2. True Negative (TN) : Total number of correct negative predictions

3. False Positive (FP) : Total number of incorrect positive predictions
4. False Negative (FN) : Total number of incorrect negative predictions

And from the components above can be used for evaluation, such as:

1. Accuracy
   To measure correct predictions
   $$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (6)$$
2. Precision

   To measure model accuracy in predicting positive class.
   $$Precision = \frac{TP}{TP + FP} \quad (7)$$
3. Recall
   To help the model measure the number of positive cases in data.
   $$Recall = \frac{TP}{TP + TN} \quad (8)$$

4. F1-Score
   Is the harmonic mean between precision and recall, used for cases where balance between both metrics is needed.
   $$f1 - Score = 2 * \frac{Precision * Recall}{Precision * Recall} \quad (9)$$

2.1.4. Random Forest Algorithm

Random Forest is an ensemble algorithm composed of multiple decision trees. It makes predictions based on the aggregated output of each individual tree [16]. The more trees included in the forest, the higher the potential accuracy and the better the model is at preventing overfitting. The working process of Random Forest consists of two main phases:

1. Combining *N* decision trees to build a Random Forest
2. Making predictions using each tree constructed in the first phase

In this study, Random Forest is implemented with the following parameters: n_estimators = 100, max_depth = None, and random_state = 42. The model is trained using the same features as the Decision Tree, which include a combination of TF-IDF, time, and event location.

2.2. Dataset Source

This study uses a dataset obtained from an official government website, BNPB (Indonesian National Disaster Management Authority), which is publicly available [3][14] and contains a total of 15,863 records. The dataset was selected based on the completeness of the information and its credibility, as it originates from an official government institution. Several criteria were used in the data collection process:

1. The dataset must include: date of occurrence, location of the incident, cause of the event, and type of disaster.
2. The data is collected from the years 2020 to 2024.
3. The dataset is verified to ensure there are no missing values or duplicated records.

Table 1. Disaster Data

Date of incident	Type of Incident	Regency	Province	Cause
31/12/2024	Flood	Pasuruan City	East Java	Moderate intensity rain....
31/12/2024	Landslide	Surakarta City	Central Java	Triggered by heavy rain ......
.....	.....	.....	.....	.....

2.3. Data Preprocessing

Data Preprocessing is a crucial initial step in this research, as it ensures that the data can be properly understood by the machine learning model, specifically the Decision Tree algorithm used in this study [4]. This stage involves data cleaning, feature transformation, and data format standardization. There are three stages of data preprocessing, namely:

2.3.1. Date Feature Transformation

This feature involves converting date values into numerical form so they can be processed by machine learning algorithms. The transformation extracts key components such as:

1. Months
   The month data can be used to analyze seasonal patterns. For example, the dry season typically starts in April and lasts until September, while the rainy season begins in October and continues through November.

2. Date
   The date data will be transformed into the day of the week and represented in a numerical format. For

example: Monday = 1, Tuesday = 2, and so on until Sunday = 7.

2.3.2. Location Encoding

Location feature encoding is a process that converts categorical data into numerical values based on alphabetical order, allowing machine learning algorithms to process the data effectively [4].

Table 2. Encoded Location Data

Regency	Province	Encode regency	Encoded province
Pasuruan	East Java	2	2
Bandung	West Java	1	1

2.3.3. Preprocessing the cause column

The "cause" column in the dataset contains text that describes the cause of a particular disaster [6]. Before this data is used in the present study, a cleaning and feature extraction process is carried out to convert the text into a numerical form that can be understood by machine learning algorithms. This process consists of three steps, namely:

1. Removing stopwords, punctuation, and numbers. For example: “yang”, “dengan”, “?”, “!”, “1”, and “2”.
2. Converting words to their root form using Natural Language Processing (NLP) for example: “Mengalirkan” is reduced to “alir”, and “bergetar” to “getar”.
3. Transforming the text in the "cause" column into TF-IDF vectors.

Table 3. Example of transformation into TF-IDF

Word	Province
Rain	0.63
River	0.52
Forest	0.32

2.4. Model Formation

This section explains how the features and target are derived from the provided data, and describes how the Decision Tree algorithm is utilized so that the model can be trained to classify types of disasters [7].

2.4.1. Features (X)

Features (X) are the numerical representations of attributes used to predict the type of disaster event. The following are the features that will be used in this study:
1. Day/Season
2. Regency and Provinsi
3. TF-IDF of the cause column

2.4.2. Target (Y)

The target (Y) refers to the data contained in the column for the type of disaster event. The target is multiclass in nature, as it consists of multiple categories, such as:
1. Flood
2. Landslide
3. Forest fire
4. Others

2.4.3. Decision Tree Algorithm

The Decision Tree algorithm is used to classify disaster types based on the processed numerical features, using a tree-like structure [11]. This algorithm builds a decision tree by recursively splitting the data based on feature values until an optimal decision is reached. There are three main parameters used, namely:

1. Criterion
   The criterion is used to determine the best split at each node of the tree. There are two common options: Gini Impurity and Entropy.
2. Max Depth
   Max Depth is used to define the maximum depth of the tree. Its purpose is to prevent the model from overfitting.
3. Min Samples Split
   Min Samples Split is used to define the minimum number of samples required to further split an internal node. Its purpose is to avoid splitting nodes based on too few data points and to help maintain the model's generalization ability.

The visualization of the decision tree structure and the analysis of important features are presented in the Results and Discussion section to support the interpretability of the model.

2.5. Model Evaluation

Model evaluation is applied to measure how accurately the Decision Tree performs in classifying types of disaster events [12][15]. This evaluation aims to ensure that the model not only performs well on the training data, but also possesses good generalization capability on unseen data.

2.5.1. Data Splitting

The dataset will be divided into two parts:
1. 80% for the training set, which is used to train the model.
2. 20% for the testing set, which contains unseen data to evaluate the model's performance.

2.5.2. Evaluation Matrix

After the model is trained and tested, its performance is evaluated using the following evaluation metrics [15]:
1. Confusion Matrix
   The Confusion Matrix is used to provide an overview of the distribution of errors and predictions for each class.

2. Accuracy
   Accuracy is used to measure the proportion of correct predictions out of the total test data.

3. Precision
   Precision is used to measure the accuracy of the model in predicting a specific class.

4. Recall
   Recall is used to indicate how well the model captures all actual instances of a specific class.

5. F1-Score
   F1-Score is used to evaluate the model by ensuring a balance between precision and recall in classification.

2.5.3. Model Performance Comparison

To evaluate the classification performance of disaster event types, this study compares two machine learning algorithms: Decision Tree and Random Forest. Both models are trained using numerical features derived from TF-IDF transformation of the *cause* column, along with location and time of the event as additional features. The comparison is conducted using evaluation metrics including: Accuracy, Precision, Recall, and F1-Score.

2.6. Tools and Library

This section explains the tools used in the research, including the software and libraries employed to test the disaster event classification model using the Decision Tree algorithm.

2.6.2. Programming Language

The programming language used is Python. Python is widely known as a popular and powerful language in the field of machine learning [11]. It also offers several libraries that are well-suited for implementing the Decision Tree algorithm and performing data processing.

2.6.3. Library Python

In this study, several Python libraries were used to support the processes of data processing, classification modeling, and performance evaluation. The libraries used are as follows:
1. Scikit-learn was used for classification using Decision Tree and Random Forest, Evaluation and TF-IDF.
2. Pandas used for managing tabular data.
3. Sastrawi used for stemming Indonesia text.
4. Mathplotlib/seaborn used for visualizing results such as the confusion matrix and evaluation score charts.

3. Results and Discussion

3.1. Data Preprocessing Results

The cleaned data consists of thousands of valid disaster events free from missing values and duplicates. It is the result of a preprocessing pipeline that includes the following steps:

1. Transformation of date into day of the week (Monday–Sunday) and conversion into season (rainy or dry).
2. Label encoding and one-hot encoding applied to the location columns (regency and province) after data cleaning.
3. Text cleaning in the “cause” column: removing stopwords, punctuation, and numbers, followed by stemming using the Sastrawi library.
4. Feature extraction using TF-IDF produces a numerical representation of the cause of the event.

Visualization of the top 10 words with the highest TF-IDF values

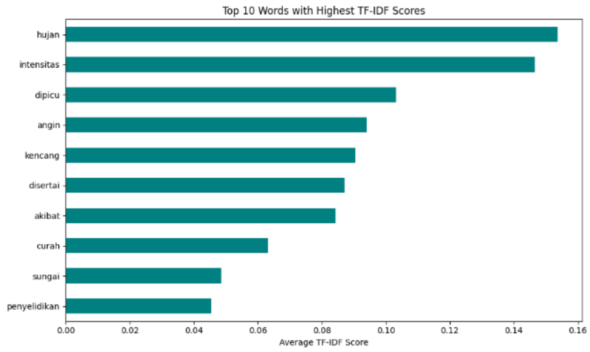

Figure 1. Top 10 Words with the Highest TF-IDF Values

Figure 1 shows the top 10 words with the highest TF-IDF values. The word *"hujan"* (rain) has the most dominant value, followed by *"intensitas"* (intensity), *"dipicu"* (triggered), and *"angin"* (wind) in the second to fourth positions. These words indicate that extreme weather and water-related conditions are major contributors to disaster events. This pattern reinforces that the causes of disasters in the dataset are strongly associated with climatological factors, particularly floods and extreme weather.

3.2. Decision Tree Evaluation

The Decision Tree model was trained using 80% of the data as the training set and 20% as the testing set. The training process involved features such as season, event location, and TF-IDF extracted features from the textual data in the cause column.

3.2.1. Model Accuracy

The model achieved an accuracy of 88%, indicating that it can recognize the types of disaster events fairly well on previously unseen data.

3.2.2. Macro Average Results

The table below shows the evaluation metric scores for each disaster class:

Table 4. Evaluation Metrics Results of the Decision Tree

Class	Precision	Recall	F1-Score	Support
Flood	0.91	0.88	0.89	80
Extreme Weather	0.85	0.89	0.83	45
Landslide	0.88	0.84	0.86	50
Forest and Land Fire	0.90	0.97	0.93	447
AVG Macro	0.66	0.58	0.60	3173
AVG Weighted	0.87	0.88	0.87	3173

The evaluation metrics in Table 4 show that the Decision Tree model performs very well in classifying majority categories such as floods, landslides, extreme weather, and forest fires, with F1-scores close to or above 0.88. This high performance indicates that the model works optimally for categories with large data distributions and more consistent causal patterns.

In contrast, classes such as volcanic eruptions, tidal waves, and droughts are not presented in the table due to an insufficient number of instances and the model's complete failure to predict them. As a result, the F1-score for these classes is 0.60.

Since the macro average F1-score is calculated as the unweighted mean of the F1-scores across all classes (without considering class support), the model's inability to predict minority classes significantly impacts the overall score, which is only 0.60.

This condition indicates that the Decision Tree model is biased toward majority classes. Therefore, strategies such as data balancing (e.g., oversampling) or the use of alternative algorithms like Random Forest could serve as potential solutions to improve performance on minority classes.

3.2.3. Macro Average Chart

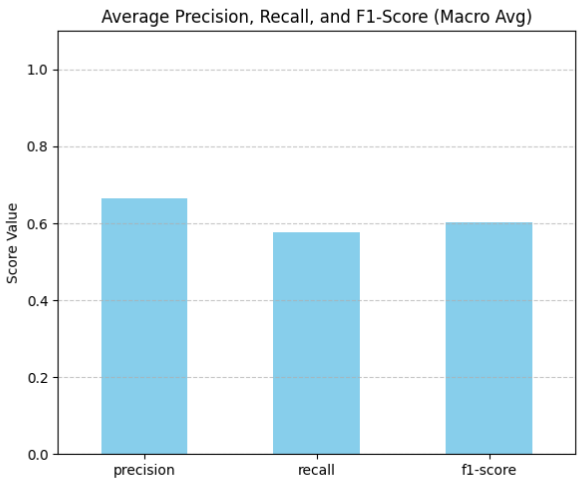

Figure 2. Macro Average Score Chart

Figure 2 shows the macro average values of the evaluation metrics for the Decision Tree model, with Precision at 0.66, Recall at 0.58, and F1-Score at 0.60. These values indicate that the model performs reasonably well in correctly identifying the relevant labels (Precision), but still lacks the ability to detect all categories comprehensively (Recall).

This gap is primarily caused by the imbalanced data distribution, where minority categories such as volcanic eruptions, which appeared only once in the test set are not adequately represented by the model.

3.2.4. Confusion Matrix Results

According to the Confusion Matrix, most of the model's predictions are correct. However, there are some misclassifications, particularly between extreme weather and floods, which are often caused by similar factors such as "heavy rain".

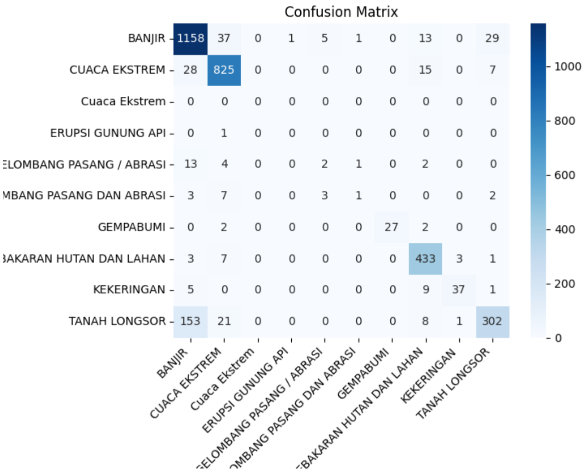

Figure 3. Confusion Matrix Model Decision Tree

Figure 3 presents the confusion matrix resulting from the evaluation of the Decision Tree model. It can be observed that most of the model's predictions are correct, as indicated by the dark blue diagonal particularly for the Banjir (Flood) class (*1,158 correct predictions*) and Cuaca ekstrem (Extreme Weather) (*825 correct predictions*). However, there is a significant misclassification in the Tanah longsor (Landslide) class, with many instances incorrectly classified as Banjir (Flood), totaling 153 cases. This error is likely due to the similarity in causes, such as Hujan tinggi (Heavy rainfall) and Aliran air pemukaan (Surface water runoff), which are commonly associated with both types of disasters. Such misclassifications highlight the challenge the model faces in distinguishing between classes that share semantically similar causes.

3.2.5. Decision Tree Visualization

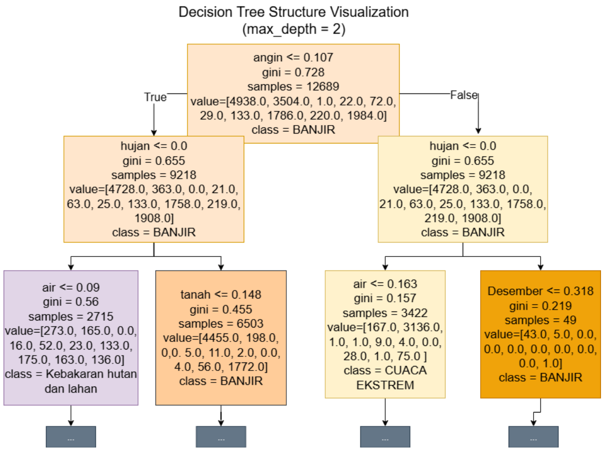

Figure 4. Decision Tree Visualization

The Decision Tree structure visualization in Figure 4 illustrates how the model makes classification decisions based on feature values derived from the TF-IDF process. The model is limited to a depth of two (*max_depth = 2*) to maintain readability, especially when included in documents or scientific reports. The first node (root) splits the data based on the feature "angin" (wind) with a threshold of ≤ 0.107. If the feature value meets this condition (True), the data is directed to the left branch; otherwise (False), it is directed to the right". Each subsequent branch continues to split based on other important features such as "hujan" (rain), "air" (water), "tanah" (soil), and "desember" (December).

At each node, the gini value (a measure of impurity), sample count (number of instances at that node), value (class distribution), and the majority class (class) predicted by the model at that node are displayed. For example, in the left branch of the root node, if the word "hujan" does not appear (≤ 0.0), the data is further split based on the word "air", which ultimately leads to a majority prediction of Forest and Land Fire. On the right branch, features like "banjir" (flood) and "desember" play a key role in classifying events as Extreme Weather or Flood.

This visualization demonstrates that certain words such as "angin", "hujan", and "banjir" are highly influential in the classification decision. By examining the tree structure, we can understand how the model maps the relationship between disaster causes (in words) and the event types. Additionally, this visualization helps evaluate whether the model's learning aligns with logical patterns relevant to disaster knowledge domains.

3.3. Random Forest Evaluation Results

3.3.1. Evaluation Metrics Results of Random Forest

The table below presents the evaluation metric values for the Random Forest algorithm across each disaster category:

Table 5. Evaluation Metrics Results of Random Forest

Class	Precision	Recall	F1-Score	Support
Flood	0.86	0.93	0.90	1244
Extreme Weather	0.92	0.96	0.94	875
Landslide	0.88	0.65	0.75	485
Forest and Land Fires	0.96	0.98	0.97	447
Macro Average	0.67	0.61	0.62	3173
Weighted Average	0.89	0.89	0.89	3173

The evaluation metrics in Table 5 indicate that the Random Forest model performs very well in classifying the majority categories such as flood, extreme weather, landslide, and forest and land fires. These four categories recorded high F1-scores, approaching or exceeding 0.90, which demonstrates that the model can effectively recognize data patterns in categories with large amounts of data and consistent causal characteristics.

However, the table does not include minority categories such as volcanic eruption, tidal wave, or earthquake, because these classes have a very small number of samples (support < 10) and were not correctly predicted by the model. The absence of correct predictions in these minority classes results in F1-scores of 0.00, and including them in the table would affect its readability.

This is clearly reflected in the macro average F1-score, which is only 0.62, despite the weighted average F1-score reaching 0.89. This indicates that the model is still biased toward majority classes, as macro average calculates the average across all classes regardless of their sample sizes.

Thus, although the *Random Forest* model performs better than the Decision Tree, it still faces challenges in handling data imbalance. Therefore, approaches such as oversampling or exploring other ensemble models are recommended to improve performance for categories with minority data.

3.3.2. Visualization of the Confusion Matrix

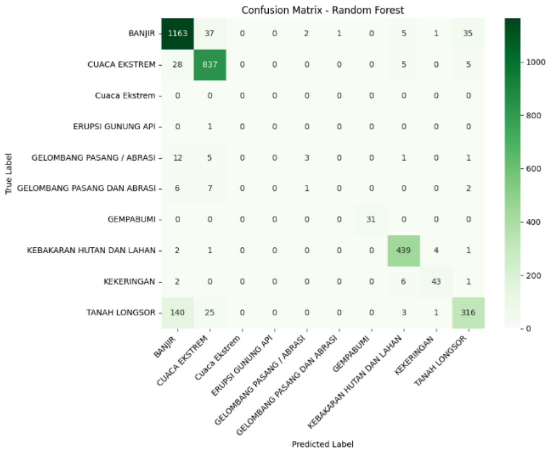

Figure 5. Confusion Matrix of the Random Forest Model

Figure 5 shows the confusion matrix resulting from the evaluation of the Random Forest model. It can be seen that most of the model's predictions are correct, as indicated by the dark green colors along the diagonal, especially in the Flood class (1,163 correct predictions) and Extreme Weather class (837 correct predictions).

However, there is a considerable misclassification in the Landslide class, where a number of instances were incorrectly classified as Flood, amounting to 140 cases. This misclassification is most likely caused by the similarity in causal patterns, such as heavy rainfall and surface runoff, which commonly occur in both types of disasters.

In addition, several classes with small amounts of data, such as Volcanic Eruption, Tidal Wave, and Tidal Wave and Abrasion, were not predicted accurately by the model. This indicates that minority classes remain a challenge, and the model's performance tends to be less optimal for classes with low data representation.

3.3.3. Graph of Macro Average Evaluation Metrics

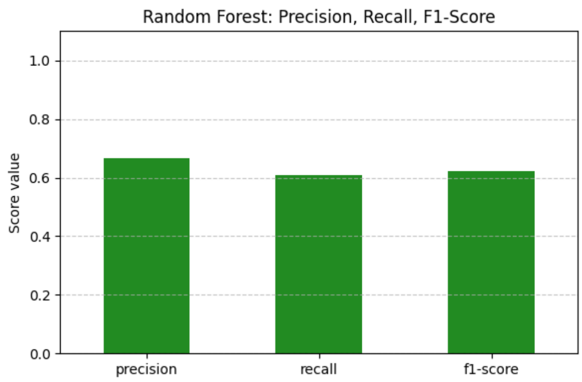

Figure 6. Macro Average Metrics Graph of Random Forest

Figure 6 shows the macro average values of the evaluation metrics for the Random Forest model, with a precision of 0.67, recall of 0.61, and F1-score of 0.62. These values indicate that although the model is fairly capable of identifying the correct labels (precision), it still struggles to detect all categories comprehensively (recall). The gap between precision and recall is caused by data imbalance within the dataset, especially in minority classes that have very few data points or even only one in the test set. As a result, the model is unable to effectively learn the patterns of these classes during training, which negatively impacts the recall and F1-score in the overall evaluation.

3.3.4. Random Forest Visualization

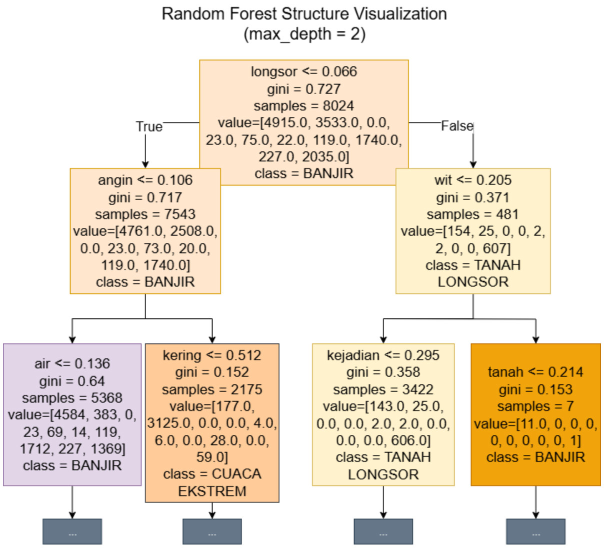

Figure 7. Random Forest Visualization

The figure above illustrates the visualization of one of the trees in the Random Forest model, simplified with a maximum depth of two (max_depth=2). Although Random Forest consists of numerous decision trees combined to produce the final prediction, the visualization of a single tree can provide insight into the basic logic used in the decision-making process.

The first node (the root) splits based on the feature "longsor" (landslide) with a threshold of ≤ 0.066. The left branch indicates that if the value of the "longsor" feature is less than or equal to 0.066, the model will consider subsequent features such as "angin" (wind) and "penyebab" (cause), which eventually lead to predictions for majority classes like Flood and Extreme Weather. On the other hand, if the "longsor" value exceeds the threshold, features such as "wit", "kejadian" (event), and "tanah" (soil) come into play in splitting the data, typically directing the prediction toward the Landslide class.

The Gini value displayed at each node represents the impurity or disorder of the data at that node the smaller the value, the more homogeneous the data in that node. For example, the lower-right branch leading to a prediction of the Flood class has a low Gini value, indicating a high level of confidence in that prediction.

Overall, this visualization provides an understanding of how one tree in the Random Forest builds classification rules based on important words derived from the TF-IDF process. While the Random Forest does not rely on a single tree alone, such an interpretation is useful for understanding the model's characteristics and which features contribute significantly to its decision-making.

3.4. Model Performance Comparison

This study compares the performance of two classification algorithms, namely Decision Tree and Random Forest, in classifying types of disaster events based on numerical and text features that have been converted using TF-IDF. The comparison is conducted based on evaluation metrics such as Accuracy, Precision, Recall, and F1-Score.

3.4.1. Accuracy

The Decision Tree model achieved an accuracy of 88%, while the Random Forest model showed a slightly higher accuracy of 89%. Although the difference is not very large, Random Forest still demonstrates more consistent and stable performance in predicting disaster event types overall.

3.4.2. Macro Average Metrics

To understand the overall performance of both models across all classes regardless of the number of data in each class, the macro average evaluation metrics are used. These metrics provide an average representation of the model's performance across all classes in a balanced way.

Table 6 below presents a comparison of the macro average precision, recall, and F1-score values between the Decision Tree and Random Forest models:

Table 6. Comparison of Random Forest and Decision Tree

Metric	Decision Tree	Random Forest
Precision	0.66	0.67
Recall	0.58	0.61
F1-score	0.60	0.62

Table 6 presents a comparison of the macro average evaluation metrics between Decision Tree and Random

Forest models, including precision, recall, and F1-score. The Random Forest model recorded a macro precision of 0.67, recall of 0.61, and F1-score of 0.62, slightly outperforming the Decision Tree, which achieved a macro precision of 0.66, recall of 0.58, and F1-score of 0.60.

The improvement in Random Forest indicates that this model is more effective in handling class imbalance. This is because the ensemble approach of Random Forest can compensate for the classification weaknesses commonly found in minority classes within the Decision Tree model.

Thus, Random Forest provides more balanced predictions across all classes, not only focusing on the majority class, resulting in more stable overall performance.

3.4.3. Confusion Matrix

Figure 3 shows the confusion matrix for the Decision Tree model, where significant misclassifications are observed between the *Landslide* and *Flood* classes. A total of 153 landslide cases were misclassified as floods, likely due to similar causes such as heavy rainfall or surface runoff. Meanwhile, *Volcanic Eruption* and *Tidal Wave* events could not be predicted at all due to the very small number of instances, which prevented the model from recognizing patterns in those categories.

In contrast, Figure 4 displays the confusion matrix for the Random Forest model, which shows a significant improvement. This model successfully reduced misclassification errors and increased correct predictions for majority classes such as *Flood* and *Extreme Weather*. However, challenges in classifying minority classes still remain due to imbalanced data distribution.

Overall, the Random Forest model proves to be more effective than the Decision Tree in:

1. Reducing errors among majority classes.
2. Improving prediction stability and accuracy.
3. Consistently achieving higher evaluation metrics (Precision, Recall, F1-Score).

However, both models still require further development to handle minority classes better, such as through data balancing techniques or more advanced ensemble methods.

3.4.4. Robustness to Imbalanced Data

One of the main challenges in disaster type classification is the uneven distribution of data. Some categories, such as Flood and Extreme Weather, have large amounts of data, while others like Volcanic Eruption and Tidal Wave only appear a few times in the dataset.

The Decision Tree model shows weaknesses in dealing with this condition. This is reflected in the low F1-score values for minority classes and high misclassification rates. This weakness occurs because Decision Trees tend to build structures favoring majority classes, making them less sensitive to patterns from rarely occurring categories.

On the other hand, the Random Forest model demonstrates better robustness to data imbalance. Although not perfect, the model is able to reduce bias toward majority classes and produce more stable predictions for minority classes. This is possible because Random Forest combines many decision trees, making it more robust in recognizing even rare data patterns.

However, both Decision Tree and Random Forest still have limitations in classifying categories with very small amounts of data. Therefore, data balancing techniques such as oversampling, undersampling, or the Synthetic Minority Over-sampling Technique (SMOTE) can be applied in future research to improve model performance on minority classes.

4. Conclusion

Based on the evaluation results, the Decision Tree algorithm achieved an accuracy of 88% with a macro F1-score of 0.60. Although this model was able to classify majority categories such as Flood, Extreme Weather, and Landslide well, its performance significantly declined on minority classes such as Volcanic Eruption and Tidal Wave, which had very limited data. This indicates that Decision Tree has weaknesses in handling imbalanced data distribution.

As an alternative, the Random Forest algorithm was tested and achieved an accuracy of 89%, with an improved macro F1-score of 0.68. Random Forest provided more stable prediction results across various disaster categories and demonstrated better robustness against data imbalance. The visualization of the confusion matrix supports these findings, showing reduced misclassification errors in several previously difficult-to-recognize classes.

Overall, Random Forest has proven to deliver more reliable classification performance compared to Decision Tree, especially in the context of uneven disaster data distribution. These results indicate that model selection and data balancing techniques are critical aspects in developing an accurate and adaptive disaster classification system.

References

[1] Baharuddin, F., & Tjahyanto, A. (2022). Peningkatan Performa Klasifikasi Machine Learning Melalui Perbandingan Metode Machine Learning dan Peningkatan Dataset. *Jurnal Sisfokom (Sistem Informasi Dan Komputer)*, *11*(1), 25–31. https://doi.org/10.32736/sisfokom.v11i1.1337

[2] Rahman, I. F., Shestia, W. A., Rama, S. D., Azzahra, S., Dermawan, A. A., Studi, P., Rekayasa, M., & Industri, F. T. (2024). Klasifikasi Diagnosa Pasien Di Klinik Sri Dengan Metode Decision Tree. *Jurnal Teknik Ibnu Sina)*, *9*(01). https://doi.org/10.36352/jt-ibsi.v8xx

[3] Rosyida, A. ; dkk. (2021). Data Bencana Indonesia 2021. *Kepala BNPB Sekretaris*, *5*(3), 248–253.

[4] Ariantoro, A., & Anggaryani, M. (2023). Pengembangan Kartu Fisika Stop-Go (Fikaso) Pada Materi Gelombang Cahaya untuk Meningkatkan Hasil Belajar Peserta Didik. *PENDIPA Journal of Science Education*, *7*(2), 312–319. https://doi.org/10.33369/pendipa.7.2.312-319

[5] Manurung, S. R., Saputra, R. E., & Setianingsih, C. (n.d.). KLASIFIKASI MODEL DECESION TREE UNTUK DETEKSI GEMPA BUMI DALAM SISTEM PERINGATAN. In *TELKATIKA* (Vol. 3, Issue 2).

[6] Septiani, D., & Isabela, I. (n.d.). *SINTESIA: Jurnal Sistem dan Teknologi Informasi Indonesia ANALISIS TERM FREQUENCY INVERSE DOCUMENT FREQUENCY (TF-IDF) DALAM TEMU KEMBALI INFORMASI PADA DOKUMEN TEKS.*

[7] Quinlan, J. R. (1986). Induction of Decision Trees. *Machine Learning*, *1*(1). https://doi.org/10.1023/A:1022643204877

[8] Ni, J., Cai, Y., Tang, G., & Xie, Y. (2021). Collaborative filtering recommendation algorithm based on TF-IDF and user characteristics. *Applied Sciences (Switzerland)*, *11*(20). https://doi.org/10.3390/app11209554

[9] Sul, S., & Cho, S. B. (2024). Understanding people's attitudes in IoT systems using wellness probes and TF-IDF data analysis. *Multimedia Tools and Applications*, *83*(35). https://doi.org/10.1007/s11042-024-18830-8

[10] Lan, F. (2022). Research on Text Similarity Measurement Hybrid Algorithm with Term Semantic Information and TF-IDF Method. *Advances in Multimedia*, *2022*. https://doi.org/10.1155/2022/7923262

[11] Jijo, B. T., & Abdulazeez, A. M. (2021). Classification Based on Decision Tree Algorithm for Machine Learning. *Journal of Applied Science and Technology Trends*, *2*(1). https://doi.org/10.38094/jastt20165

[12] Song, Y. Y., & Lu, Y. (2015). Decision tree methods: applications for classification and prediction. *Shanghai archives of psychiatry*.

[13] Qaiser, S., & Ali, R. (2018). Text mining: use of TF-IDF to examine the relevance of words to documents. *International journal of computer applications*, *181*(1), 25-29.

[14] Azizah, M. A., Khoirudin Apriadi, R., Tri Januarti, R., Winugroho, T., Yulianto, S., Kurniawan, W., & Dewa Ketut Kerta Widana, I. (2021). Kajian risiko bencana berdasarkan jumlah kejadian dan dampak bencana di indonesia periode tahun 2010–2020. *PENDIPA Journal of Science Education*, *6*(1), 35-40.

[15] Yacouby, R., & Axman, D. (2020). *Probabilistic Extension of Precision, Recall, and F1 Score for More Thorough Evaluation of Classification Models*. https://doi.org/10.18653/v1/2020.eval4nlp-1.9

[16] Binus University, 2024. Random Forest vs Decision Tree. [Online] (Updated 2 Apr 2024) Tersedia di: https://sis.binus.ac.id/2024/04/02/random-forest-vs-decision-tree/ [Accessed 5 Aug 2025].