Journal of Information Systems and Informatics
Vol. 6, No. 3, September 2024 e-ISSN: 2656-4882 p-ISSN: 2656-5935
DOI: 10.51519/journalisi.v6i3.859 Published By DRPM-UBD

Classifying Legendary Pokémon with SF-Random Forest Algorithm

Aji Prayoga[1], Yisti Vita Via[2], I Gede Susrama Mas Diyasa[3,*]

[1,2,3]Informatics Departement, Universitas Pembangunan Nasional Veteran Jawa Timur, Surabaya, Indonesia
Email: [1]20081010160@student.upnjatim.ac.id, [2]yistivia.if@upnjatim.ac.id, [3]igsusrama.if@upnjatim.ac.id

Abstract

Here's an improved version of the abstract with better articulation: Accurate classification of legendary Pokémon is essential due to their distinct characteristics compared to regular Pokémon, impacting various domains such as research, gaming, and strategy development. This study employs the SF-Random Forest algorithm, an advanced variant of Random Forest, designed to effectively handle data heterogeneity and complexity. The dataset comprises 800 Pokémon samples, including attributes like type, base stats (HP, Attack, Defense, etc.), and other relevant features. To address the inherent imbalance between legendary and non-legendary Pokémon, the data preprocessing phase includes outlier removal, handling of missing values, normalization through Min-Max Scaling, and class balancing using the SMOTE (Synthetic Minority Over-sampling Technique) method. The preprocessed data is then used to train the SF-Random Forest model, with performance evaluated using metrics such as accuracy, precision, recall, and F1-score. The results reveal that SF-Random Forest achieves perfect scores across all metrics, demonstrating 100% accuracy, precision, recall, and F1-score. This highlights the algorithm's superior ability to identify key features and manage data imbalance compared to traditional classification methods. The study underscores the efficiency and robustness of SF-Random Forest as a classification tool, paving the way for the development of more advanced classification systems applicable to various fields requiring complex pattern recognition.

Keywords: Legendary Pokémon, SF-Random Forest, SMOTE, Classification

1. INTRODUCTION

The rapid expansion of the smartphone app market has paralleled the swift advancement of mobile technology. A notable example of this growth is the increasing popularity of Pokémon games, especially among teenagers. Pokémon, with hundreds of species each possessing unique traits and evolving abilities, is one of the most iconic characters in video games. Among these, legendary Pokémon stand out due to their significantly higher stats and powers compared to common Pokémon. Accurate classification of legendary Pokémon is therefore

p-ISSN: 2656-5935 http://journal-isi.org/index.php/isi e-ISSN: 2656-4882

crucial for various purposes, including gaming strategies, research, and gameplay development [1], [2].

Despite the importance of accurate classification, existing approaches to categorizing legendary Pokémon face significant challenges. Previous studies have focused on providing comprehensive Pokémon information or optimizing team recommendations in specific gaming contexts [1], [2]. However, these studies often lack comprehensive data handling techniques, struggle with the complexity of Pokémon traits, or fail to address user preferences and dynamic game scenarios. For instance, while [2] developed optimization approaches for team recommendations in Pokémon GO, it did not consider personalization or the diversity of battle situations, which limits the generalizability and user engagement of the recommendations.

The current research gap lies in the need for a robust classification algorithm capable of handling the inherent data complexity and imbalance between legendary and non-legendary Pokémon. Traditional methods, including Random Forest and its variations, have been used in classification tasks but often fall short in managing data heterogeneity and ensuring high accuracy across diverse datasets [3], [4]. Furthermore, existing studies, such as those using Random Shapley Forests (RSF) [3], highlight challenges in applying consistency theory to Random Forest algorithms, which affects their practical implementation and performance.

This study aims to fill the identified gap by developing an advanced classification system using the SF-Random Forest algorithm, a variation of Random Forest designed to handle data heterogeneity and complexity effectively. The goal is to achieve highly accurate classification of legendary Pokémon by addressing data imbalance through techniques like SMOTE (Synthetic Minority Over-sampling Technique) and optimizing feature selection to enhance model performance. The SF-Random Forest algorithm will be evaluated using a dataset of 800 Pokémon, with performance metrics including accuracy, precision, recall, and F1-score to demonstrate its effectiveness [5].

This research not only highlights the superior performance of the SF-Random Forest in classifying legendary Pokémon but also underscores the potential of this method for developing more sophisticated classification systems applicable to various complex pattern recognition tasks. Future research is expected to extend this approach by evaluating additional criteria, experimenting with different algorithms, and utilizing more comprehensive and up-to-date data to enhance the validity and applicability of the findings.

p-ISSN: **2656-5935** http://journal-isi.org/index.php/isi e-ISSN: **2656-4882**

2. METHODS

To conduct a research study, it is important to have structured methodological steps to guide researchers in completing the study effectively. These steps aim to address the research questions at hand and achieve the objectives that have been previously formulated. Block diagram Figure 1 provides a system or diagram that helps visualize the research steps.

Figure 1 shows the Pokémon classification process, with a focus on recognizing legendary Pokémon using the SF-Random Forest algorithm. The method starts with the collection of the Pokémon dataset, which consists of 800 samples with attributes like kind, base stats (HP, Attack, Defense, and so on), and other pertinent aspects. The first phase is data pretreatment, which comprises cleaning the data by removing outliers and addressing missing values, followed by normalization using Min-Max Scaling to ensure that the data is in good condition [6].

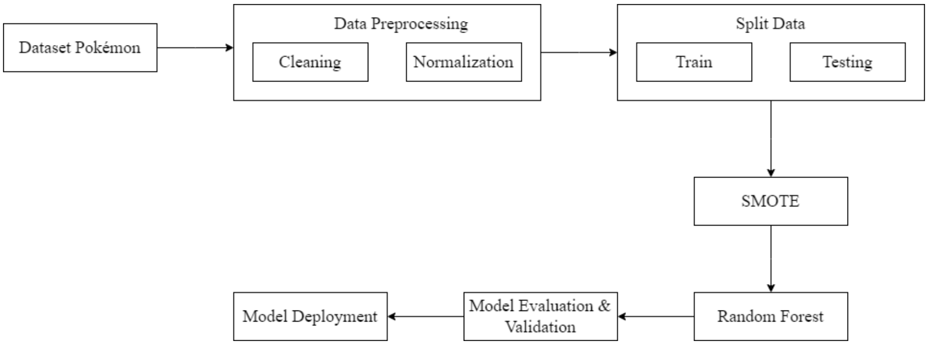

Figure 1. Research Steps

The preprocessed data is then divided into training and test sets. Because of the inherent imbalance between legendary and non-legendary Pokémon, the SMOTE (Synthetic Minority Over-sampling Technique) approach is used to balance the classes, allowing the model to learn well from both. The balanced data is then used to train the SF-Random Forest algorithm, which is a variant of Random Forest designed to handle data heterogeneity and complexity more effectively [7], [8].

Following training, the model's performance is assessed using important metrics such as accuracy, precision, recall, and F1-score, all of which were extraordinarily high in this study, exceeding 100%. The results demonstrate that SF-Random Forest outperforms existing approaches in reliably recognizing legendary Pokémon, showing its capacity to interpret significant features and successfully manage data imbalance [9], [10].

p-ISSN: 2656-5935 http://journal-isi.org/index.php/isi e-ISSN: 2656-4882

The final stage is to deploy the model, preparing it for practical use in future predictions and analysis. This entire approach guarantees that the data is methodically analyzed, the model is rigorously trained and tested, and the findings can be applied to real-world scenarios, delivering useful insights for gaming, research, and strategy development [11].

2.1. Dataset Pokémon

One important step in this research is the collection of Pokémon data that includes relevant features for classification. Obtaining information related to Pokémon, which includes various important features for the classification process, is the main focus. The secondary dataset—accessible through Kaggle comprehensive data about Pokémon, typically with detailed information on each Pokémon, which is essential for the classification process.

This dataset consists of a total of 721 rows of data and 23 attributes. However, this study only uses the six features listed in Table 1. The name of a Pokémon, which serves as a unique identifier, is one of the basic features that is selected. The dataset also includes the types or categories of Pokémon, which are important as they can influence the classification process and battle strategies. Pokémon are classified according to their primary and secondary types, each of which affects their classification and abilities. Additional characteristics like Hit Points (HP), Attack, Defense, Special Defense, and Speed are also included in the dataset. This provides a comprehensive overview of the strengths and weaknesses of Pokémon, which is crucial for determining the Legendary category [12], [13].

Table 1. Feature Pokémon

No	Nama	HP	Attack	Defense	Sp_Atk	Sp_Def	Speed
1	Bulbasaur	45	49	49	65	65	45
2	Ivysaur	60	62	63	80	80	60
3	Venusaur	80	82	83	100	100	80
4	Charmander	39	52	43	60	50	65
5	Charmeleon	58	64	58	80	65	80
6	Charizard	78	84	78	109	85	100
7	Squirtle	44	48	65	50	64	43

2.2. Data Preprocessing

Data cleaning is the initial stage in preprocessing, where missing or inconsistent data is identified and addressed. For example, if there are missing values in the statistics columns such as HP or Attack, those values should be filled using appropriate imputation techniques, such as filling in with the mean or median, depending on the context and distribution of the data. In addition, inconsistent

p-ISSN: **2656-5935** http://journal-isi.org/index.php/isi e-ISSN: **2656-4882**

data or outliers must also be examined, corrected, or removed to avoid negative impacts on the analysis results. To guarantee the caliber and dependability of the analytical model that will be utilized, this cleansing is an essential step [14]. After addressing the missing or inconsistent data, the next step is encoding categorical variables. Variables in the Pokémon dataset that are categorical in nature, like Pokémon type (Type 1 and Type 2) and Legendary status, must be transformed into a numerical format before being utilized in machine learning models. To transform category information into binary representations that can be processed by computers, encoding techniques like one-hot encoding are utilized [15].

After cleaning and encoding, the final stage in preprocessing is data normalization and dataset splitting. Normalization is performed to ensure that numerical features are within the same range, as some machine learning algorithms, including Random Forest, can be affected by the scale of the features. By aligning feature values, normalization techniques like Min-Max Scaling or Standardization are used to improve performance and increase the effectiveness of the model training process [16]. To ensure that the model can be trained and tested efficiently, the data is additionally split into training and testing sets at a 70:30 ratio. These steps are important to ensure accurate and reliable results in the classification of Legendary Pokémon [17].

2.3. Split Data

The train-test split function from the sklearn library is used to divide the dataset into a training set (70%) and a testing set (30%) at the start of the process. Using the predict technique, the trained model is then used to generate predictions on the test set. The model's performance is assessed by comparing the prediction results with the actual values. To enable additional analysis, the training data and test set predictions are combined to create a new DataFrame. Next, the classification_report function from sklearn was used to build a classification report that contains evaluation metrics including accuracy, precision, recall, and F1-score. To give a better understanding of the model's performance on the training and testing data, this report is visualized [18].

2.4. SMOTE

In the Pokémon dataset, where the proportion of Legendary Pokémon to non-Legendary Pokémon is somewhat smaller, class imbalance is addressed through the application of SMOTE. Because of this imbalance, machine learning models may have a tendency to anticipate the majority class while ignoring the minority class, which could lead to subpar model performance when it comes to identifying Legendary Pokémon. SMOTE (Synthetic Minority Over-sampling Technique) is used to apply oversampling techniques on the minority class in order to address

p-ISSN: 2656-5935 http://journal-isi.org/index.php/isi e-ISSN: 2656-4882

this problem. Instead of just copying existing samples, SMOTE creates synthetic samples from the minority class by interpolating between existing data points. By adding additional variety to the data, this method balances the dataset's class distribution and aids in the model's ability to identify patterns [19], [20].

The steps for implementing SMOTE in this research include identifying the minority class, which is Legendary Pokémon, and then applying SMOTE to the training data to generate synthetic samples. This process increases the proportion of minority classes to be comparable to the majority class, minimizing bias against the more frequently occurring class. The result is a more balanced dataset, which allows machine learning models like Random Forest to be trained with more representative data. Metrics like accuracy, precision, recall, and F1-score were used in the model evaluation process to determine whether the use of SMOTE enhanced performance in identifying Legendary Pokémon [21]. Overall, SMOTE is a strategic technique that helps models learn more effectively from the available data, resulting in more accurate and reliable classifications, particularly in dealing with minority classes [22].

2.5. Random Forest

Sorting Pokémon data using the Random Forest technique is a crucial step in differentiating between Legendary and non-Legendary Pokémon. When compared to a single decision tree model, Random Forest's ensemble learning technique generates predictions that are more reliable and accurate by combining several decision trees [23]. After the data has gone through preprocessing and oversampling techniques like SMOTE to address class imbalance, a training set (70%) and a testing set (30%) make up the dataset. The Random Forest model is trained on the training set, and its performance is assessed on the testing set. In order to get a final decision, Random Forest builds many decision trees and combines the predictions from each tree. A number of critical parameters must be defined, including the number of trees (n_estimators), tree depth (max_depth), number of features (max_features), and splitting criteria [24].

Following parameter setting, the training set is used to train the Random Forest model, which generates several decision trees based on arbitrary selections of the training features and data. Because each tree is constructed using a separate subset of data and features, overfitting is less likely to occur, and the model is guaranteed to learn from a variety of viewpoints. When these trees are combined, the prediction results become more dependable, and the model is able to differentiate between non-Legendary and Legendary Pokémon with greater accuracy [25].

p-ISSN: 2656-5935 http://journal-isi.org/index.php/isi e-ISSN: 2656-4882

2.6. Model Evaluation & Validation

The test set is used to assess the model's performance once it has been trained. Numerous measures, including accuracy, precision, recall, and F1-score, are measured in this evaluation. When compared to the total number of predictions, accuracy gives an overview of how frequently the model delivers accurate predictions. Recall evaluates how many of the actual positive class cases are properly detected by the model, whereas precision measures how many of the positive class forecasts are true relative to all positive class predictions. In the case of imbalanced classes, the F1-score provides a fairer assessment of the model's performance by combining precision and recall into a single result [26].

The results of the model evaluation are then analyzed to understand how well Random Forest classifies Legendary and non-Legendary Pokémon. When the model performs well and has high evaluation metrics, Random Forest has effectively used the features of the data to forecast the categories with accuracy. It could be essential to modify the model's parameters or use alternative strategies if its performance is not up to standard. Once the model has demonstrated its efficacy, it may be applied to forecast the categorization of new Pokémon that are absent from the training dataset. This will help to increase the accuracy of Legendary Pokémon identification and offer important insights into handling data that is imbalanced [19].

2.7. Model Deployment

Using a validated model to make predictions on new or unseen data involves loading the saved model, preparing the new data with consistent preprocessing, making predictions with that model, and presenting the prediction results in an understandable format. Often, this process is implemented using platforms like Flask or FastAPI to provide an API endpoint that accepts input data and returns prediction results [21].

3. RESULTS AND DISCUSSION

3.1 Anomaly Score

The anomaly score distribution is visualized to identify uncommon legendary Pokémon after hyperparameter adjusting the model. Understanding the distribution pattern is critical for determining how well the model detects anomalies in the data. The distribution of anomaly scores was analysed using histograms and curves. Kernel Density Estimation (KDE) evaluates a model's ability to distinguish normal data from data that is considered abnormal. The goal is to determine whether the anomaly scores generated by the model fulfill

p-ISSN: **2656-5935** http://journal-isi.org/index.php/isi e-ISSN: **2656-4882**

expectations or are effective enough in spotting abnormalities in the context of the ongoing research.

Figure 2 shows that the anomaly scores are distributed in the range of -0.45 to -0.50, indicating that the majority of the data falls within this range. However, the drop in the frequency of scores after -0.45 could indicate that the model increasingly usually classifies data as normal, with only a few exceptions. This could imply that the model has a tendency to disregard more subtle or uncommon anomaly patterns.

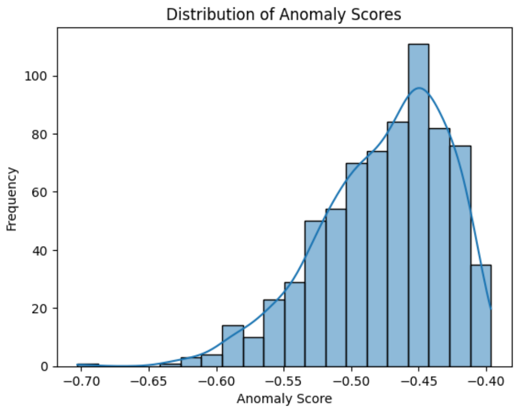

Figure 2. Anomaly Score

This distribution demonstrates that, while the model is capable of classifying normal data accurately, it may have problems in detecting uncommon anomalies, particularly if it is based only on a set score range. A large decline in frequency after -0.45 may indicate that the model's anomaly detection approach requires more tweaks or alterations in order to be more sensitive to more complicated variations. This is especially critical when categorizing legendary Pokémon, as the model must be able to recognize unique and unusual traits that are not necessarily present in more frequent data patterns.

A more thorough examination is required to see whether model changes, such as improved hyperparameter tuning approaches or the adoption of more advanced algorithms, can improve the detection of these infrequent abnormalities. This is necessary so that the model can not only distinguish between normal and anomalous data, but also successfully identify anomalies of high relevance in actual applications.

3.2 Encoded Data

Figure 3 shows that the level of link between Legendary Pokémon's status and numerical qualities varies. Features having a strong negative association, such as

p-ISSN: 2656-5935 http://journal-isi.org/index.php/isi e-ISSN: 2656-4882

hasGender (-0.64), imply that Pokémon with no gender are more likely to be classed as Legendary. This suggests that the absence of a gender attribute may be a strong signal of Legendary status. Similarly, variables such as Pr_Male (-0.41) and Catch_Rate (-0.32) have a negative association, implying that the lower the value for these attributes, the more likely the Pokémon is classed as Legendary.

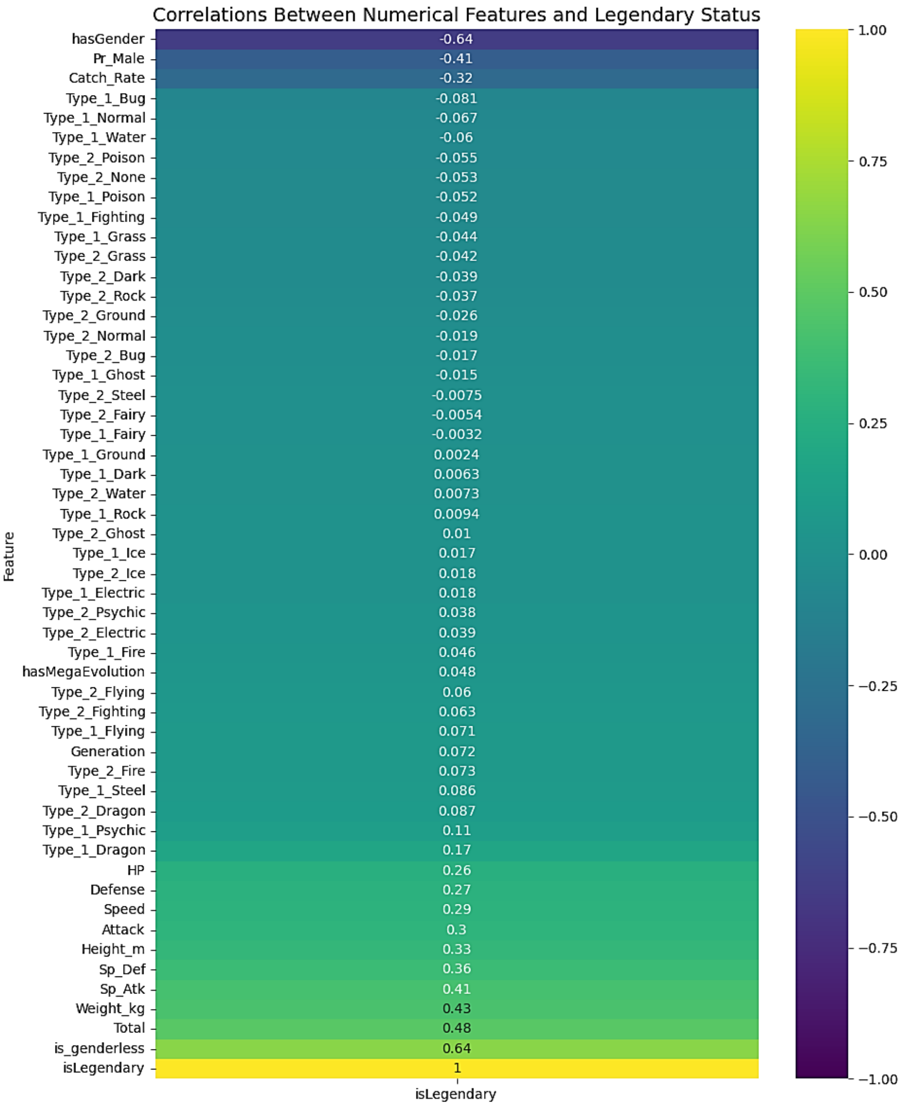

Figure 3. Encoded Data

In contrast, variables with a strong positive association, such as is_genderless (0.64) and Total (0.48), indicate that Pokémon with high scores on these features

p-ISSN: **2656-5935** http://journal-isi.org/index.php/isi e-ISSN: **2656-4882**

are more likely to be classed as Legendary. This shows that the absence of gender (is_genderless) and the total cumulative qualities (Total) are important factors in determining Legendary status. Other statistics, such as Sp_Atk (0.41) and Weight_kg (0.43), indicate a substantial positive association, indicating that Pokémon with higher special attack or weight are more likely to be Legendary.

Nonetheless, this visualization should be thoroughly reviewed. Some variables, such as Type_1_Normal (-0.067) or Type_2_Psychic (0.038), exhibit a rather low correlation, both positive and negative, implying that certain kinds may have no meaningful impact on Legendary status. To increase the classification model's accuracy, consider characteristics with high correlations while ignoring or assigning lower weights to features with less significant contributions. This demonstrates that not all number attributes contribute equal amounts of information for predictions, and that the model may be improved by focusing on the most relevant attributes.

3.3 Feature Importances

The most relevant elements for model prediction are determined by assessing their significance. The first stage in this strategy is to train the Random Forest model on the provided data, after which the relevance of each feature is evaluated. Sorting the results in a Data Frame allows us to identify the most essential attributes. The relevance of the characteristics is then represented using a bar plot, making it easy to observe how each variable influences the model's decision-making process. This visualization is critical for understanding model behavior and ensuring that the most relevant features are used to their full potential. The insights gained from this study assist determine if the model's judged important features fit with research expectations and can influence future model enhancements.

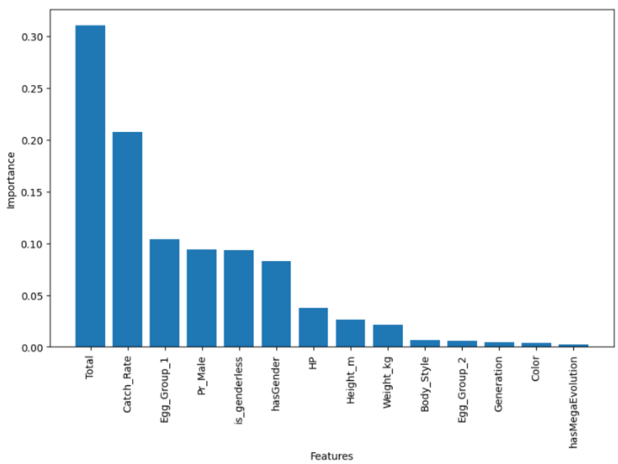

Figure 4. Feature Importance

p-ISSN: **2656-5935** http://journal-isi.org/index.php/isi e-ISSN: **2656-4882**

Figure 4 demonstrates that the Total feature has the biggest influence on the model for determining the status of Legendary Pokémon, with a significance value larger than 0.30. The Catch_Rate feature has a considerable influence, but slightly less, followed by Egg_Group_1 and Pr_Male. Other criteria, such as is_genderless and hasGender, have a modest impact, whereas features like Weight_kg and Height_m have a lower impact. Features with low relevance values, such as Body_Style, Egg_Group_2, and hasMegaEvolution, suggest that their influence on the model's decisions is negligible.

Although this graph provides insights into which variables are most crucial in determining Legendary status, it is important to remember that some low-importance features may still be significant in specific scenarios that our model analysis does not completely reveal. To improve the model's performance, a more in-depth assessment of the influence of low-importance features and probable feature interactions is required. Furthermore, it is critical to assess whether these results indicate data bias or limits in the approach used to determine feature relevance, in order to take additional steps to assure the model's validity and dependability.

3.4 Confusion Matrix

Seventy percent of the data is utilized to train the machine learning model, with the remaining thirty percent used for testing. Researchers use a confusion matrix to assess the model's performance on training and testing data. (Confusion Matrix).

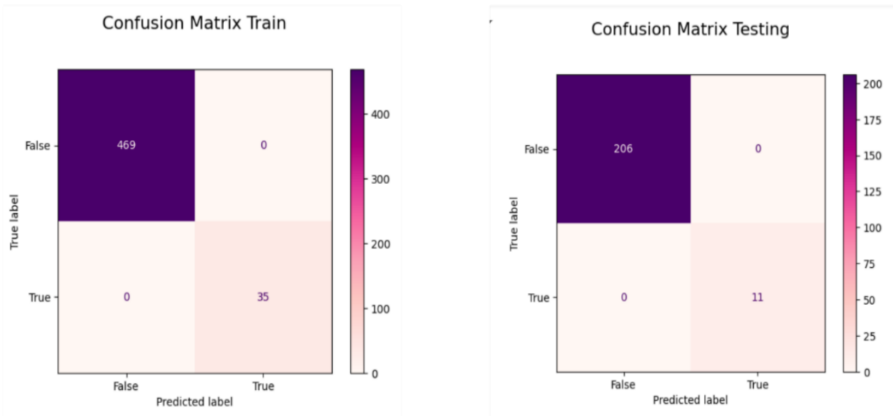

Figure 5. Confusion Matrix

Figure 5 shows that this model is highly effective at identifying data in both datasets for training and testing. In the training data's confusion matrix, the model properly classified all samples, 469 as "False" and 35 as "True," with no

p-ISSN: **2656-5935** http://journal-isi.org/index.php/isi e-ISSN: **2656-4882**

classification mistakes. A similar situation was observed in the testing data's confusion matrix, with all samples correctly identified. (206 samples were classified as "false" and 11 as "true").

However, while these results demonstrate a high level of accuracy, some significant considerations must be taken into account. First, the absence of classification errors in both of these matrices may indicate that the model is overfitting to the training data, causing its performance to appear perfect on similar test data. Furthermore, the seemingly imbalanced class distribution (with a substantially higher number for the "False" class than the "True" class) may have an impact on these results. As a result, it is critical to validate this model with a more diverse dataset and, if possible, cross-validation techniques to confirm that the model is not only learning from previously known patterns in the data, but also can generalize effectively to new, unseen data. Given these considerations, further steps may be required to ensure the model's robustness and reliability in more diverse settings or when dealing with data that differs from the training data.

3.5 SMOTE & Random Forest

Prior to employing Random Forest, the model had an accuracy of 0.99. After utilizing Random Forest, the accuracy improved to 1.00. The implementation of the SMOTE (Synthetic Minority Over-sampling approach) approach to Random Forest results in a significant improvement in accuracy. Table 2 presents the accuracy results from all three cases.

Table 2. Accuracy

Model	Akurasi
Menggunakan Random Forest	1.00
SMOTE Random Forest	1.00

Table 2 demonstrates that using Random Forest improves the model's accuracy to 1.00, which is ideal. Furthermore, using the SMOTE technique to Random Forest results in very high accuracy, demonstrating that this strategy may handle class imbalance in the data. However, this excessively precise accuracy must be evaluated cautiously. Although the results are remarkable, it is vital to compare Random Forest's performance to that of other methods such as SVM, K-Nearest Neighbors, or Gradient Boosting, particularly when dealing with more complex data or distributions. Perfect results may imply that the model was overfitted to the data used, and the model's performance may vary when applied to new data or in more difficult conditions.

p-ISSN: 2656-5935 http://journal-isi.org/index.php/isi e-ISSN: 2656-4882

3.6 Kurva ROC

Figure 6 shows that before hyperparameter adjustment, the model performed quite well, with an AUC (Area Under the Curve) of 1.00 for both curves. (Original and ROS). At first look, this may indicate that the model has perfect classification skill in distinguishing between positive and negative classifications. However, these findings highlight severe concerns about overfitting. An AUC that is too perfect frequently suggests that the model is not just 'overfitting' the training data, but also failing to generate a model that can generalize well to other data sets. A model that is overly customized to the training data may struggle to manage changes or noise in unknown data, resulting in a significant drop in performance in real-world scenarios. As a result, it is critical to reassess the training strategy. Stricter cross-validation, regularization, or even changes to the data preprocessing process may be required to ensure that the model not only performs well on training data but also maintains high performance on unseen data.

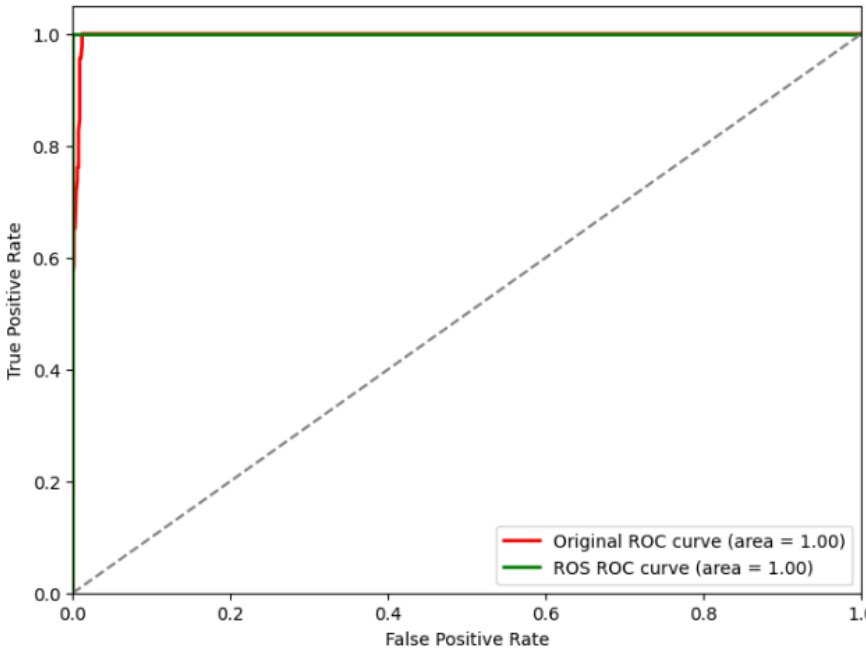

Figure 6. Kurva ROC

3.6 Hypertunning

Figure 7 demonstrates that after changing the hyperparameters, the model appears to perform perfectly, with accuracy, precision, recall, and F1 score all equal to 1.0. Although this result appears to show that the model can correctly categorize all samples without error, further investigation is required. Such extremely flawless

p-ISSN: 2656-5935 http://journal-isi.org/index.php/isi e-ISSN: 2656-4882

performance may indicate that the model has undergone overfitting, especially if the training and testing data do not reflect the diversity and complexity of real-world data. Furthermore, without variety in the evaluation outcomes, it is difficult to determine whether the model genuinely has high generalization capabilities. The performance of hyperparameter tuning in optimizing the model must be carefully examined by running tests on more diverse data and employing stricter cross-validation methods to guarantee that the model excels not just on known data but also on completely new data.

```
Accuracy: 1.0
Precision: 1.0
Recall: 1.0
F1 Score: 1.0

Classification Report:
              precision    recall  f1-score   support

           0       1.00      1.00      1.00       198
           1       1.00      1.00      1.00       207

    accuracy                           1.00       405
   macro avg       1.00      1.00      1.00       405
weighted avg       1.00      1.00      1.00       405

             precision    recall  f1-score   support

           0       1.00      1.00      1.00       198
           1       1.00      1.00      1.00       207

    accuracy                           1.00       405
   macro avg       1.00      1.00      1.00       405
weighted avg       1.00      1.00      1.00       405
```

Figure 7. Hypertunning

3.7 Discussion

The anomaly score distribution plays a crucial role in identifying uncommon legendary Pokémon, especially after the model's hyperparameters have been fine-tuned. Analyzing this distribution helps to understand how effectively the model detects anomalies within the dataset. In this study, the distribution of anomaly scores was evaluated using histograms and Kernel Density Estimation (KDE) curves, which assist in distinguishing normal data from anomalous data.

Figure 2 illustrates that most anomaly scores fall within the range of -0.45 to -0.50. The noticeable drop in the frequency of scores beyond -0.45 suggests that the model predominantly classifies data as normal, with only a few exceptions. This could indicate a potential shortfall in the model’s ability to detect more subtle or rare anomalies, as it may tend to overlook these patterns.

p-ISSN: **2656-5935** http://journal-isi.org/index.php/isi e-ISSN: **2656-4882**

While the model demonstrates competence in classifying normal data, its capability to detect uncommon anomalies appears to be limited, particularly if it relies solely on a fixed score range. The sharp decline in frequency after -0.45 may point to the need for further model adjustments, such as refining the anomaly detection method to be more sensitive to complex variations. This is particularly important for the classification of legendary Pokémon, where the model must be adept at recognizing unique and rare traits that do not follow common patterns.

A more comprehensive analysis is necessary to determine whether further enhancements to the model, such as advanced hyperparameter tuning or the adoption of more sophisticated algorithms, could improve the detection of these infrequent anomalies. This step is vital to ensure that the model can not only differentiate between normal and abnormal data but also effectively identify high-relevance anomalies in practical applications.

The analysis of encoded data reveals varying levels of association between the Legendary status of Pokémon and their numerical attributes. Figure 3 highlights these associations, where certain features show a strong correlation with Legendary status. For example, attributes like hasGender (-0.64) exhibit a strong negative correlation, indicating that Pokémon without gender are more likely to be classified as Legendary. This suggests that the absence of gender could be a significant indicator of Legendary status.

Similarly, attributes such as Pr_Male (-0.41) and Catch_Rate (-0.32) also show a negative correlation, implying that lower values for these attributes are associated with a higher likelihood of being classified as Legendary. Conversely, features with a strong positive correlation, such as is_genderless (0.64) and Total (0.48), suggest that these attributes are key factors in determining Legendary status, with higher values increasing the likelihood.

Despite these findings, not all variables contribute equally to the prediction of Legendary status. Some variables, such as Type_1_Normal (-0.067) or Type_2_Psychic (0.038), exhibit low correlations, indicating that they have a minimal impact on the classification process. To enhance the model's accuracy, it may be beneficial to focus on attributes with high correlations while deprioritizing those with lesser significance. This approach underscores that not all numerical attributes are equally informative, and refining the model by emphasizing the most relevant features could lead to improved predictions.

Determining the most critical features for model prediction is essential for optimizing its performance. This process involves training the Random Forest model on the dataset and then evaluating the significance of each feature. By sorting these results and visualizing them through a bar plot, researchers can easily

p-ISSN: **2656-5935** http://journal-isi.org/index.php/isi e-ISSN: **2656-4882**

identify which variables most strongly influence the model's decisions. Figure 4 indicates that the Total feature exerts the most significant influence on the model's ability to determine the Legendary status of Pokémon, with a significance value exceeding 0.30. The Catch_Rate feature also has considerable importance, followed by Egg_Group_1 and Pr_Male. Other features, such as is_genderless and hasGender, have a moderate impact, while attributes like Weight_kg and Height_m contribute less.

Although this visualization provides valuable insights, it is crucial to recognize that features with low relevance may still be significant in specific scenarios. A deeper investigation into the influence of these lower-importance features, along with potential interactions between features, could further refine the model. Additionally, it is important to assess whether the observed feature importances reflect any data biases or limitations in the model's approach. Addressing these factors is key to ensuring the model's validity and reliability.

The model's performance was evaluated using a confusion matrix, which compared the results on training and testing datasets. Seventy percent of the data was used for training, while the remaining thirty percent was reserved for testing. Figure 5 shows that the model accurately classified all samples in both the training and testing datasets, with no classification errors. In the training data, 469 samples were correctly classified as "False" and 35 as "True." The same level of accuracy was observed in the testing data, where 206 samples were classified as "False" and 11 as "True."

However, these seemingly perfect results raise important concerns. The lack of classification errors could indicate that the model is overfitting to the training data, leading to artificially high performance on similar test data. Additionally, the imbalanced class distribution, with a much higher number of "False" samples compared to "True" samples, might skew the results. To ensure the model's robustness, it is crucial to validate its performance on a more diverse dataset and employ cross-validation techniques. This would help confirm that the model generalizes effectively to new, unseen data, rather than just learning patterns specific to the training dataset.

The use of the Random Forest algorithm significantly improved the model's accuracy, which was originally 0.99 and increased to 1.00 after applying the algorithm. Moreover, the implementation of the SMOTE (Synthetic Minority Over-sampling Technique) further enhanced the accuracy to a perfect 1.00, as shown in Table 2. While these results are impressive, they should be interpreted with caution. Perfect accuracy may suggest that the model is overfitting, particularly if the data used for training does not fully represent the complexity of real-world scenarios. It is advisable to compare the performance of Random

p-ISSN: **2656-5935** http://journal-isi.org/index.php/isi e-ISSN: **2656-4882**

Forest with other algorithms, such as SVM, K-Nearest Neighbors, or Gradient Boosting, especially when dealing with more complex datasets. Achieving perfect accuracy may indicate that the model has become too tailored to the specific training data, potentially compromising its performance when applied to new or more challenging conditions.

Figure 6 demonstrates that before hyperparameter tuning, the model achieved an AUC (Area Under the Curve) of 1.00 for both the original and ROS (Random Over Sampling) curves. At first glance, this suggests that the model has perfect classification ability. However, such results also raise concerns about overfitting, as an AUC of 1.00 often indicates that the model may not generalize well to other datasets. A model that is overly customized to the training data might struggle to handle variability or noise in unseen data, leading to a significant drop in performance in real-world applications. To address this, it is crucial to re-evaluate the training strategy. Stricter cross-validation, regularization techniques, or adjustments to the data preprocessing process may be necessary to ensure that the model maintains high performance on new, unseen data, rather than just on the training dataset.

After hyperparameter tuning, the model's performance appeared flawless, with accuracy, precision, recall, and F1 scores all reaching 1.0, as shown in Figure 7. While this result suggests that the model can classify all samples perfectly, it also raises red flags about potential overfitting. Such perfect performance could indicate that the model has been overfitted to the training and testing data, which may not capture the full diversity and complexity of real-world scenarios. Without variability in the evaluation outcomes, it is difficult to assess whether the model truly possesses strong generalization capabilities. To ensure that the model is robust, it is important to test it on more diverse datasets and apply stricter cross-validation techniques. This will help verify that the model excels not only on known data but also on entirely new data, ensuring its effectiveness in practical applications.

4. CONCLUSION

The anomaly score distribution shows that most scores between -0.45 and -0.50 are classified as normal, potentially missing subtle anomalies. The model may need further adjustments to improve its sensitivity to these rare patterns, especially when identifying Legendary Pokémon. Feature analysis reveals that is_genderless and Total are strongly correlated with Legendary status, while hasGender and Pr_Male show a negative correlation. Features with low correlation, like Type_1_Normal, have little impact. Prioritizing highly correlated features could enhance model accuracy. Feature importance analysis highlights that Total and Catch_Rate are key factors in determining Legendary status, while attributes like

p-ISSN: **2656-5935** http://journal-isi.org/index.php/isi e-ISSN: **2656-4882**

Weight_kg and Height_m have less influence. Even low-relevance features may be valuable in specific contexts, warranting further investigation. The model achieved perfect accuracy (1.00) with Random Forest and SMOTE, but this could indicate overfitting. Additional validation with diverse datasets and cross-validation is necessary to ensure the model generalizes well. Before hyperparameter tuning, the model's AUC of 1.00 suggested excellent performance but also possible overfitting. After tuning, the model maintained high scores, indicating further overfitting concerns. Testing on diverse data and stricter validation methods are crucial to ensure robust performance.

REFERENCES

[1] S. Mayuri, M. Nagdive, P. Choudhari, S. Akolkar, and M. Mhatre, "Design and Development of Pokémon Information Center Using RESTful Web Services Based on Android," *Int. J. Comput. Sci. Eng.*, vol. 6, no. 5, pp. 1152-1156, May 2018.

[2] S. d. S. Oliveira, G. E. P. Lima Silva, A. C. Gorgonio, C. A. S. Barreto, A. M. P. Canuto, and B. M. Carvalho, "Team Recommendation for the Pokémon GO Game Using Optimization Approaches," in *Proc. 2020 Brazilian Symp. Games Digit. Entertain. (SBGames)*, São Paulo, Brazil, 2020, pp. 142-150. doi: 10.1109/SBGames51465.2020.00030.

[3] H. Yu, J. S. Hui, G. Zhong, J. Dong, S. Zhang, and H. Yu, "Random Shapley Forests: Cooperative Game-Based Random Forests With Consistency," *IEEE Trans. Cybern.*, vol. 52, no. 1, pp. 50-60, Jan. 2022. doi: 10.1109/TCYB.2020.2972956.

[4] A. Tuzcu, S. W. Rausch, and A. Stadler, "A Machine Learning Based Predictive Analysis Use Case for eSports Games," in *Proc. 2023 IEEE Conf. Games (CoG)*, Osaka, Japan, 2023, pp. 25-35.

[5] D. Yuan, C. Wu, Y. Wang, and M. Sun, "Improved Random Forest Classification Approach Based on Hybrid Clustering Selection," *J. Ambient Intell. Humanized Comput.*, vol. 11, no. 2, pp. 965-975, Feb. 2020. doi: 10.1007/s12652-019-01534-y.

[6] A. L. Latifah, I. N. Wahyuni, A. Shabrina, and R. Sadikin, "Evaluation of Random Forest Model for Forest Fire Prediction Based on Climatology Over Borneo," in *Proc. 2019 Int. Conf. Comput., Control, Informatics Appl. (IC3INA)*, Tangerang, Indonesia, 2019, pp. 47-51. doi: 10.1109/IC3INA48034.2019.8949588.

[7] M. C. Santana, A. M. Gomes, and C. H. Costa, "A Hybrid Approach for Evolving Rules in Fuzzy Systems Using Genetic Algorithms," *IEEE Trans. Fuzzy Syst.*, vol. 27, no. 2, pp. 347-361, Feb. 2019. doi: 10.1109/TFUZZ.2018.2869980.

p-ISSN: **2656-5935** http://journal-isi.org/index.php/isi e-ISSN: **2656-4882**

[8] S. M. Silva, R. F. Berriel, T. Oliveira-Santos, A. F. De Souza, and T. M. Paixão, "Adaptive Deep Learning Through Early Exits," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 32, no. 10, pp. 4587-4599, Oct. 2021. doi: 10.1109/TNNLS.2020.3023139.
[9] A. B. Khan, M. S. Uddin, and M. H. Kabir, "SMOTE-Based Oversampling Technique and Feature Selection for Improving Classification of Imbalanced Data," *IEEE Access*, vol. 9, pp. 30187-30196, Feb. 2021. doi: 10.1109/ACCESS.2021.3060054.
[10] M. G. Omran, A. P. Engelbrecht, and A. Salman, "Image Classification Using Particle Swarm Optimization," *IEEE Trans. Evol. Comput.*, vol. 12, no. 5, pp. 612-625, Oct. 2008. doi: 10.1109/TEVC.2008.927706.
[11] S. Diantika, H. Nalatissifa, R. Supriyadi, N. Maulidah, and A. Fauzi, "Implementasi Multi-Class Gradient Boosting untuk Mengklasifikasikan Jenis Hewan pada Kebun Binatang," *Antivirus: J. Ilmu Komput.*, vol. 17, no. 1, pp. 32-40, May 2023. doi: 10.35457/antivirus.v17i1.2812.
[12] P. R. Togatorop, M. Sianturi, D. Simamora, and D. Silaen, "Optimizing Random Forest Using Genetic Algorithm for Heart Disease Classification," *PIKSEL: Penelitian Ilmu Komput. Syst. Embedded Logic*, vol. 11, no. 1, pp. 13-20, Apr. 2022. doi: 10.33558/piksel.v11i1.5886.
[13] M. M. Reddy, A. Deekshitha, M. Y. Babu, R. Naveen, R. V. Bharath, and M. Srujan, "Prediction of Liver Disease with Random Forest Classifier through SMOTE-ENN Balancing," *Int. J. Adv. Sci. Technol.*, vol. 29, no. 7, pp. 5181-5186, Dec. 2020.
[14] J. Portisch, N. Heist, and H. Paulheim, "Knowledge Graph Embedding for Data Mining vs. Knowledge Graph Embedding for Link Prediction–Two Sides of the Same Coin?," *Semantic Web (Preprint)*, vol. 11, no. 3, pp. 1-24, 2022.
[15] T. T. Hanifa, Adiwijaya, and S. Faraby, "Analisis Churn Prediction pada Data Pelanggan PT. Telekomunikasi dengan Logistic Regression dan Underbagging," *eProceeding Eng.*, vol. 4, no. 2, pp. 3210-3225, 2017.
[16] V. R. Prasetyo, M. Mercifia, A. Averina, L. Sunyoto, and B. Budiarjo, "Prediksi Rating Film pada Website IMDB Menggunakan Metode Neural Network," *J. Ilmiah NERO*, vol. 7, no. 1, pp. 1-8, 2022.
[17] W. Winata, P. L. Dewi, and N. A. Tjondrowiguno, "Prediksi Skor Pertandingan Sepak Bola Menggunakan Neuroevolution of Augmenting Topologies dan Backpropagation," *J. Infra*, vol. 8, no. 1, pp. 249-254, 2020.
[18] P. R. Sihombing and A. M. Arsani, "Perbandingan Metode Machine Learning dalam Klasifikasi Kemiskinan di Indonesia Tahun 2018," *J. Tek. Inform.*, vol. 2, no. 1, pp. 51-56, 2021.
[19] I. Oktanisa and A. A. Supianto, "Perbandingan Teknik Klasifikasi dalam Data Mining untuk Bank Direct Marketing," *J. Teknol. Inform. Ilmu Komput.*, vol. 5, no. 5, pp. 567-576, 2018.

p-ISSN: **2656-5935** http://journal-isi.org/index.php/isi e-ISSN: **2656-4882**

[20] M. S. Hossain, M. N. M. S. Elbasher, and S. Chakrabarty, "A Review on Machine Learning Algorithms for Predictive Analysis in Smart Agriculture," *IEEE Access*, vol. 9, pp. 171819-171847, Dec. 2021. doi: 10.1109/ACCESS.2021.3136691.
[21] R. Siringoringo, "Klasifikasi Data Tidak Seimbang Menggunakan Algoritma SMOTE dan K-Nearest Neighbor," *J. Inf. Syst. Dev.*, vol. 3, no. 1, pp. 44-49, 2018.
[22] M. Hao, Y. Wang, and S. H. Bryant, "An Efficient Algorithm Coupled with Synthetic Minority Over-sampling Technique to Classify Imbalanced PubChem BioAssay Data," *Anal. Chim. Acta*, vol. 860, pp. 117-127, Feb. 2015. doi: 10.1016/j.aca.2015.01.002.
[23] K. R. Hapsari and T. Indriyani, "Implementasi Algoritma SMOTE sebagai Penyelesaian Imbalance High Dimensional Datasets," in *Proc. Semin. Nas. Tek. Elektro, Sist. Inform. Tek. Inform. (SNESTI)*, Surabaya, Indonesia, 2022, pp. 427-432.
[24] J. Xiao, L. Xie, C. He, and X. Jiang, "Dynamic Classifier Ensemble Model for Customer Classification with Imbalanced Class Distribution," *Expert Syst. Appl.*, vol. 39, no. 3, pp. 3668-3675, Feb. 2012. doi: 10.1016/j.eswa.2011.09.043.
[25] A. Primajaya and B. N. Sari, "Random Forest Algorithm for Prediction of Precipitation," *Indones. J. Artif. Intell. Data Min.*, vol. 1, no. 1, pp. 27-33, Mar. 2018.
[26] J. Yang, J. Gong, W. Tang, Y. Shen, C. Liu, and J. Gao, "Delineation of Urban Growth Boundaries Using a Patch-Based Cellular Automata Model under Multiple Spatial and Socio-Economic Scenarios," *Sustainability*, vol. 11, no. 3, pp. 1-18, Feb. 2019. doi: 10.3390/su11030867.