Scientific Journal of Informatics
Vol. 11, No. 3, Aug 2024
p-ISSN 2407-7658 https://journal.unnes.ac.id/journals/sji/index e-ISSN 2460-0040

Performance Comparison of Random Forest (RF) and Classification and Regression Trees (CART) for Hotel Star Rating Prediction

Annisaa Utami[1*], Dimas Fanny Hebrasianto Permadi[2], Yesy Diah Rosita[3] , Jumanto Unjung[4]
[123]Department of Informatics, Telkom University, Indonesia
[4]Department of Computer Science, Universitas Negeri Semarang, Indonesia

Abstract.
Purpose: This study proposes to evaluate the effectiveness of Random Forest (RF) compared to Classification and Regression Trees (CART) in prediction of hotel star ratings. The objective is to identify the algorithm that provides the most reliable and accurate classification outcomes based on diverse hotel attributes in accordance with the standard categorization of star hotel categories. This is necessary due to the important role of accurate star ratings in guiding consumer choices and enhancing competitive positioning in the hospitality industry.
Method: This study conducted a comprehensive dataset about Hotel in Banyumas Regency, including location, facilities, the size of rooms, type of rooms, price of rooms, and customer reviews, subjected to training through both RF and CART algorithms. Both algorithms are evaluated using accuracy, precision, recall, and F1 score. Additionally, both algorithms due to in the same preprocessing while performing hyperparameter tuning improve the efficacy of each model.
Result: The results showed that RF achieved the best overall accuracy and robustness than CART across all tests conducted. Furthermore, RF also outperformed CART in classification effectiveness among classes, including enhanced precision and recall scores across multiple stars rating categories, signifying increased generalization and consistency in classification tasks. RF classifier consistently surpassed the CART classifier in terms of both accuracy and F1-score throughout all random states and test sizes, with a highest score of 0.9932 at a random state of 100 and a test size of 0.4. The most reliable results were obtained using RF with 42 random states and a test size of 0.2, resulting in an accuracy of 0.9909, precision of 1.0, recall of 1.0, and F1 score of 1.0. Simultaneously, CART shows values of 0.9818, 1.0, 1.0, and 1.0, respectively, while maintaining the same variation. This consistent performance, regardless of fluctuations, illustrates the robustness and suitability of RF for classification tasks compared to CART.
Novelty: This study offered new insights about the implementation of machine learning about hotel star rating predictions using RF and CART algorithms. Also, the novelty of the collected hotel dataset used in this study. A detailed comparative analysis was also provided, contributing to the existing literature by showing the effectiveness of RF over CART for this specific application. Future studies could explore the integration of additional machine learning methods to further enhance prediction accuracy and operational efficiency in the hospitality industry.

Keywords: Hotel star rating classification, Performance comparison, Random forest, Classification and regression trees
Received August 2024 / **Revised** August 2024 / **Accepted** August 2024

INTRODUCTION

Traveling is a basic need for individuals due to the usefulness in relieving fatigue as observed from the statistics showing that millennials travel for a minimum of one time in a year [1]. Therefore, Banyumas Regency provides several choices of hotels spread across different locations with various classes of accommodation, rental prices, facilities, and services, which further influence ratings [2]–[4]. The classification using star rating enhances the understanding of the differences in quality and facilities between different hotels in an area [5], [6]. Rating is important in determining the quality of customer experience with higher values observed to have the capacity to enhance attraction and trust. However, the assessment is often a complex and subjective process due to the need to rate the facilities, services, location, and price. This is essential as ratings function as an indicator of the quality and variety of services offered, aiding consumers in making informed choices. As a result, it is crucial to accurately forecast hotel star

*Corresponding author.
Email addresses: annisaau@telkomuniversity.ac.id
DOI: 10.15294/sji.v11i3.11068

ratings based on a variety of attributes in order to enhance customer satisfaction and improve the industry's competitive position [5], [7].

There are features that affect the rating of starred hotels that cause difficulties and time required in the assessment process such as categories of room size, room type, price, facilities, review comments and so on. In addition, there is the possibility of missing value data in the dataset that can add to the complexity associated with hotel rating prediction [5], [6], [8]. This situation prompted the Regulation of the Minister of Tourism and Creative Economy of the Republic of Indonesia Number PM.53/HM.001/MPEK/2013 concerning Hotel Business Standards to formulate the features needed to classify star ratings [2], [3]. Therefore, based on these problems, this research applies prediction of star hotel class classification with machine learning using the RF and CART algorithms.

RF is a tree-based method that implements recursive data division into two or more distinct groups within certain predetermined boundaries until the desired result is achieved. This method relies on defined criteria and characteristics to determine the categorization result [7], [9]–[11]. The benefits include the ability to manage missing data to maintain accuracy and efficiency in processing data across multiple features and classes [4], [12], [13]. Furthermore, RF is advantageous in the classification of hotels according to star ratings. Initially, RF can integrate a variety of input features, such as amenities, room type and size, room price, and customer reviews, without significant performance degradation. Secondly, the algorithm exhibits significant resilience to overfitting, especially when faced with inconsistent data, like divergent customer assessments and varying pricing. By averaging the outcomes from multiple trees, RF mitigates the danger of overfitting the training data, resulting in more generalized and precise predictions. Moreover, the system can proficiently handle absent values in the dataset, a prevalent issue in actual hotel data. [11], [14].

CART has features identical to those of RF. The CART algorithm also facilitates the generation of trees with many classifications exceeding two. It can also be utilized to construct regression and classification trees. The presence of a categorical data scale in the bound variable results in the creation of a classification tree. Moreover, CART can be used to link between one or more bound and independent variables [11], [15]–[17]. The algorithm has the capacity to produce the classification model required to predict the star class of a hotel based on the features provided. The trend shows the possibilities of classifying hotels using RF and CART algorithms based on the features provided, such as facilities, prices, ratings, etc. [18]–[20]

The novelty is the initiation of the study through data collection followed by the design of a feature model for the classification process through the application of RF and CART algorithms. Moreover, a previous study developed a system recommendation for the best hotel in Purwokerto, Banyumas Regency, using Simple Additive Weighting (SAW) with a focus on the location, price, stars, and facilities [3]. Another study provided a decision system for hotel selection using the Technique For Order Preference By Similarity To Ideal Solution (Topsis) based on Geographic Information System (GIS) [18]. Therefore, this study aims to determine the comparison of the effectiveness of the relative advantages of the RF and CART algorithms in the classification of hotel star ratings in Banyumas Regency. The classification algorithms were developed to predict the categorization of star hotels based on the specified features following data collection. This study considers and assesses the algorithm's efficacy by comparing performance parameters such as accuracy, precision, and recall. The results are expected to improve understanding of the implementation of various machine learning techniques in the hotel industry, thereby assisting managers and stakeholders in the optimization of customer satisfaction and service delivery.

METHODS

This section discusses the methodologies employed in the research. This includes a discussion of the data collection scheme and data preprocessing, as well as a comparison of RF and CART.

Research flowchart

Figure 1 describes the methodology for gathering hotel datasets in Banyumas Regency and analyzing important attributes that are aggregated and implemented into the RF and CART algorithms in preprocessing steps. During the preprocessing phase, the data gathered from the specified attributes, including amenities, prices, ratings, and others are cleaned and filled if there is empty or missing data for use in the next steps. Data normalization involves selecting and altering the data for computation. Data classification is to classify the star hotels utilizing both RF and CART, with the procedure noted to partition the dataset into two segments, including training and testing, at a specified ratio. Predictive analysis aims

to evaluate the efficacy of each method in terms of accuracy, precision, and recall. In addition, the results obtained by the algorithm are supported by actual star rating data for hotels in Banyumas. Assessing the statistical significance of the performance associated with both methodologies is essential to ensure the accuracy of the model.

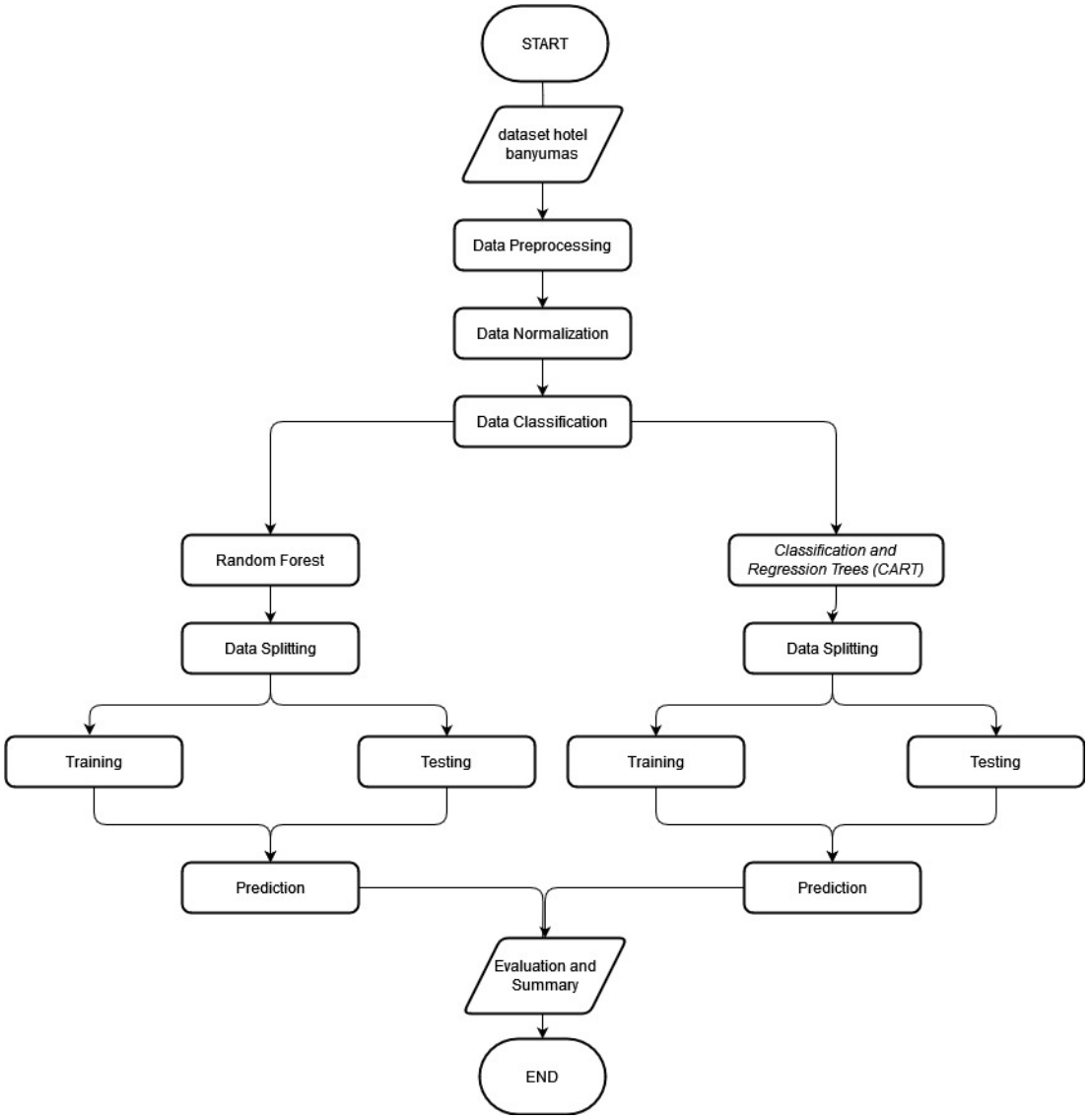

Figure 1. Classification process flowchart

Data collection

The dataset contains information about many variables related to hotels in Banyumas Regency, including room types and dimensions, amenities, room prices, and other related elements described in Table 1. The data is gathered from several sources to guarantee a thorough dataset for precise forecasts. The principal sources comprised Google Scraping, Traveloka, Ticket.com, Booking.com, and additional travel websites, with the data collection period extending from September 2023 to May 2024. The focus was on scraping hotel data information, including names, addresses, room quantities and types, amenities, pricing, and guest evaluations. Therefore, the raw dataset obtained comprises approximately 453 rows and 32 columns, detailing 184 hotel names, addresses, room counts and types, facilities, prices, guest reviews, latitude, longitude, neighborhood, phone numbers, and descriptions.

Table 1. Data description

Variable	Data_Type	Description
Name	String	the name of hotel
Rooms_Standard	numeric	number of standard rooms
Avg_Size_Standard	numeric	the average of minimum size the Standard Room
Rooms_Suite	numeric	number of suite rooms
Avg_Size_Suite	numeric	the average of minimum size the Suite Room
Hotel_Rating	String	The rating of hotel
Price_RP	numeric	The price of each rooms hotel
Room_Type	String	The type of hotel rooms

Data preprocessing

The preprocessing stage is important for preparing the dataset for effective machine learning model training. This was achieved through comprehensive cleaning to address missing values and ensure consistency across all features [12], [21]. The missing values were addressed using mean imputation, which was considered suitable for numerical features such as room sizes and prices. This was considered important because the algorithms required data without empty cells in order to avoid invalidity of the data for calculation and worse prediction results. Additionally, mode imputation was utilized for categorical variables, including room types. The normalization of the Type_Room column was noticed based on star rating and room size specifications [22], [23]. The normalization function classified each hotel room as either "Standard" or "Suite" according to defined parameters. A 2-star hotel with a room size of 20 was classified as "Standard," whereas a 4-star hotel with a room capacity of 48 was designated as "Suite." This procedure guaranteed that the dataset was thorough and prepared for subsequent analysis. Furthermore, the normalization adhered to the stipulations specified in the Hotel Criteria detailed in Table 2, which were derived from the Regulation of the Minister of Tourism and Creative Economy of Indonesia Number PM.53/HM.001/MPEK/2013 on Hotel Business Standards.

Table 2. Hotel criteria minimum standard

Rating Star	Standard Rooms	Suite Rooms	Room Size (Standard / Suite)	Facilities
1	15	0	16	In-room bathroom, no restaurant, parking
2	20	1	22	AC, TV, security, restaurant, WiFi, bar, sports, lobby, telephone, receptionist
3	30	2	24 / 48	Pool, valet parking, hot and cold bathroom, shower, bathtube, lift
4	50	3	24 / 48	Large lobby 100 m2
5	100	4	26 / 52	Lobby, rest area, public toilet, concierge staff

The dataset was systematically preprocessed to make sure the integrity and consistency were necessary for precise analysis and modeling. Initially, entries with missing hotel ratings were omitted to focus exclusively on comprehensive and valid data. Furthermore, the missing data in critical columns such as Room_Size, Type_Room, and Rooms were systematically filled in accordance with the hotel's star rating criteria. A 3-star hotel without room size information was designated a standard size of 24 square meters, and vacant Type_Room entries were assigned the default value of "Standard". The total room count was determined by aggregating the anticipated 'Standard' and 'Suite' rooms based on the hotel's star classification, so ensuring a uniform and comprehensive dataset for each hotel. This process is important for maintaining the reliability of the data set and establishing a foundation for further investigation.

The last step is augmenting the dataset through normalization and feature extraction to facilitate algorithm analysis and modeling. The normalization process involved transforming the 'Hotel_Rating' column into a numerical representation to facilitate further use in the training model and performance analysis. The dataset was augmented by aggregating room data to make sure that the 'Rooms_Standard' and 'Rooms_Suite' columns accurately represent the total counts of 'Standard' and 'Suite' rooms. At the same time, 'Avg_Size_Standard' and 'Avg_Size_Suite' indicate the average sizes of each category based on the available data. This aggregation organizes and provides the data with the accumulation of room distribution and average sizes in each hotel, enabling comparative analysis and informed decision-making. The resulting data was merged with other critical attributes, including Hotel_Rating, Facility, and Price_RP, to construct a comprehensive and consistent dataset. The efficacy of the preprocessing stages is assessed through a comparison of raw and normalized data in a table that illustrates the enhancement of the data set's usability for modeling applications.

Data classification

Random forest

The RF is an ensemble learning method for classification by employing essential metrics such as Impurity Gain and Information Gain to clarify the quality of splits in decision trees. Impurity Gain quantifies the likelihood that a randomly chosen element would be inaccurately identified when assigned labels based on the distribution of the dataset. For classification task, the output of the RF classes is selected by most trees. This can be accomplished with Equation (1) [2], [7], [9], [13], [24]:

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

Figure 2 discusses that the estimator parameters range from 1 to 300, with the model being trained on each of these values. Furthermore, each subset for p_i was created by randomly picking data points from the original dataset using the bootstrap sampling method. The approach guaranteed that each subset contained an equivalent number of data points as the original dataset, despite the possibility of repetitions or missing. A decision tree was constructed for each subgroup utilizing a randomly chosen subset of features. At each node of the tree, features were selected to partition the data according to parameters such as gain impurity or information gain. This recursive procedure persisted until a stopping requirement was met, such as achieving a maximum tree depth or a minimum number of samples at a leaf node. The depth used in this case were 10, 20, and 30, while the minimum split sample was 2, 5, and 10. The last parameter was the leaf node, which was set at 1, 2, and 4. The application for independent predictions followed the construction of all decision trees. In classification tasks, the final prediction was determined through the majority voting among the decision trees [25].

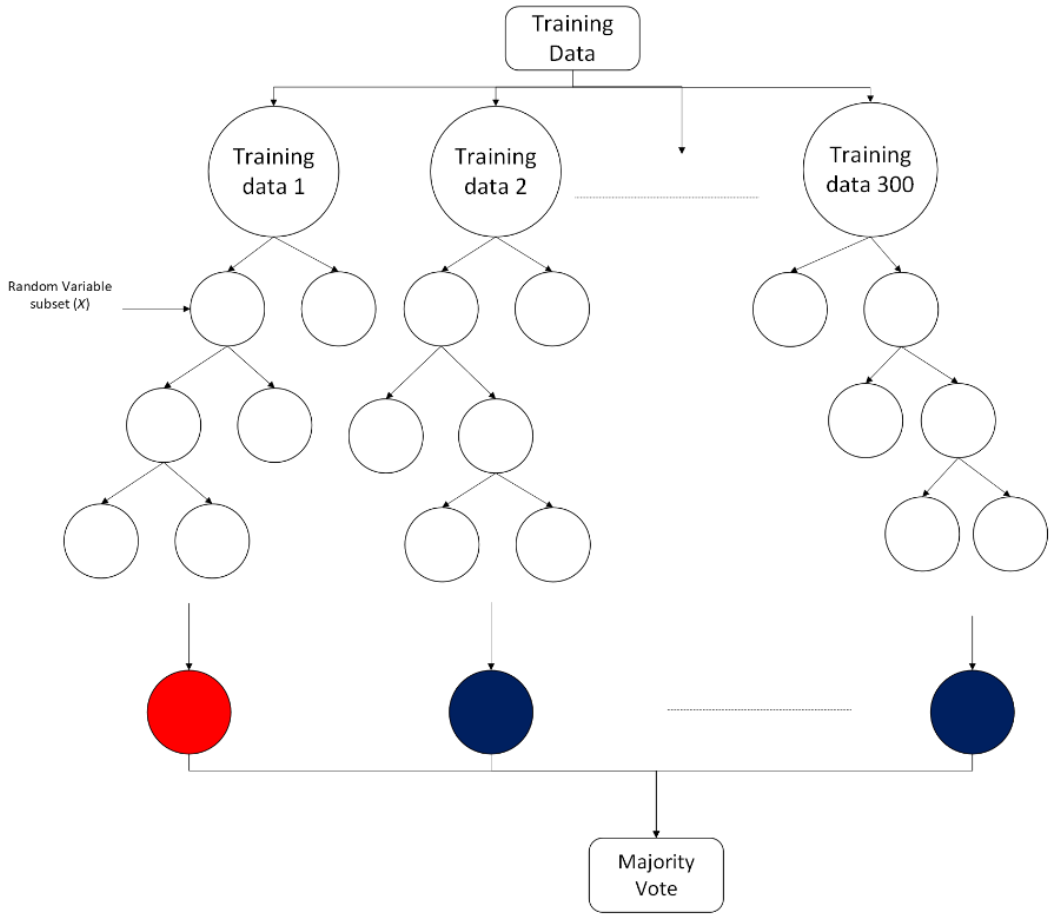

Figure 2. Random forest model [14]

Classification and regression tree (CART)

CART algorithm is simple but has good classification performance within a short computing time [16], [26]. In the training stage, CART can develop a tree model using split rules with all the training variables designated as nodes in the model tree. This can be achieved by using entropy, which focuses on finding the information gain value for each attribute to use the largest as the node. Therefore, Equation (2) can be used to determine the entropy and Equation (3) for the information gain value.

$$Entropy\ (S)\ =\ -\sum_i^c p_i \log_2 p_i \tag{2}$$

Where c is the class label on the data $p_i$is the amount of data that has class i, and c is the number of classes.

$$IG(S,a) = Entropy(S) - \sum v \frac{|S_V|}{|S|} Entropy(S_V) \tag{3}$$

Where *IG* is the value of information gain, v is all values that include the data set of attributes a. S_v is the amount of data owned by the attribute a[16], [27]. The labeling of hotel classification classes on a node can use the rule of the highest number where the class of the data is the class that has the largest value of $P(t_0|i)$, The value of $P(t_0|i)$ is obtained using the Equation (4).

$$P(t_0|i) = max_t P(t|i) = max_t \frac{N_t(i)}{N(i)} \tag{4}$$

Where $P(t|j)$ is the frequency of occurrence of class t in hotel classification at node i, $N_t(j)$ is the number of attributes that have class t at node i, and $N(i)$ is the number of records at node i [16], [23]. The steps associated with CART are presented in Figure 3, where 95 nodes and 90 leaves are used in this study.

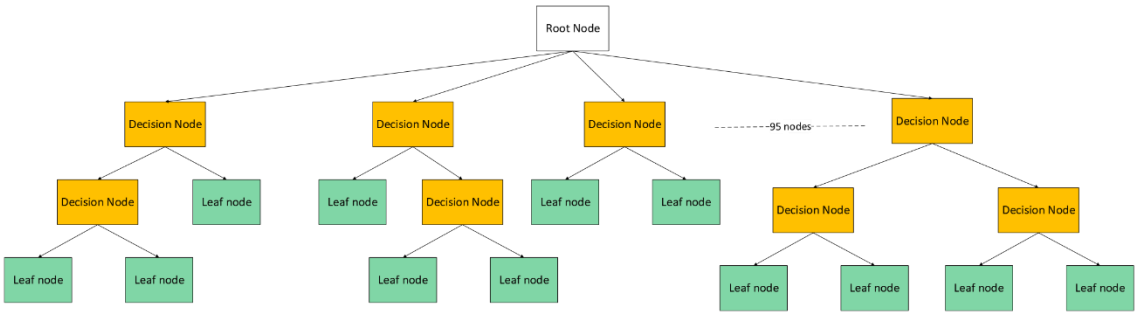

Figure 3. CART model [11], [28], [29]

Synthetic minority over-sampling technique (SMOTE)

SMOTE (Synthetic Minority Over-sampling Technique) is used to address class imbalance in the dataset. This imbalance happens when some classes, such as specific hotel star ratings, are underrepresented compared to others, which can cause the model to be biased towards the majority class and perform poorly on the minority classes. For instance, there may be more 1-star hotels than 5-star hotels in the dataset. SMOTE helps correct this imbalance by generating synthetic samples for the minority class, thus balancing the dataset and improving the model's performance across all classes. The x is a minority class instance and x_k is one of its k-nearest neighbors, a synthetic instance x'. The Equation is used to generate a new minority class as in Equation (5) [9], [22], [25].

$$x' = x + rand(0,1) \times |x - x_k| \tag{5}$$

RESULTS AND DISCUSSIONS

Data preprocessing

The sample dataset collected is presented in Table 3 with several empty cells before normalization. The application of preprocessing steps, such as normalization and feature extraction, structured the dataset for further analysis and modeling. The normalized dataset provided that the 'Rooms_Standard' and 'Rooms_Suite' columns are the counts of standard and suite rooms, respectively, which were gathered from the original 'Type_Room' and 'Rooms' columns. Additionally, 'Avg_Size_Standard' and 'Avg_Size_Suite' provided average dimensions of room categories derived from 'Room_Size'. It was also observed that 'Hotel_Rating' was changed to a numerical value from Non-Stars, Stars 1, Stars 2, Stars 3, Stars 4, Stars 5

to 0, 1, 2, 3, 4, and 5, respectively, in order to facilitate model training and evaluation. The objective was to accelerate the modeling process by verifying that the rating was presented in a uniform and quantitative format.

The missing values in the Room_Size column are filled based on the minimum classification criteria for each hotel's rating. The fill_room_size function is employed to validate and assign absent values in Room_Size, if necessary, according to ratings based on the criteria stated in hotel_criteria. Hotels classified as '3 Star' without any room size present in the dataset were filled as standard room size according to the criteria for three-star hotels. This verified that all entries had significant Room_Size values, allowing for a more precise analysis. Empty values in the Type_Room column are resolved using the fill_type_room function. For example, missing values were set to 'Standard' according to the hotel rating standards to standardize room types if there was missing information, thus obtaining more consistent results across the dataset, as shown in Table 2.

A similar approach is used to fix missing values in the Room column in the case of room types, which are filled using the fill_rooms function. The step is to obtain the total number of rooms by combining each Room_Type categorized as 'Standard' and 'Suite' based on the Hotel Criteria Minimum Standards in Table 2. This process effectively assesses each hotel's facilities by providing a detailed count of the number of available rooms. Furthermore, the 'dropna' function is used to remove rows with missing values for Hotel_Rating from the dataset to ensure that only entries with a particular rating are included in the analysis. This approach is essential to maintaining the integrity and reliability of the dataset.

By employing the aggregate_rooms_sizes function, the dataset was refined by aggregating the number of rooms and average sizes for each hotel based on the 'Standard' and 'Suite' hotel categories to improve comparative analysis and decision-making results by accumulating the total and the average size rooms. This function provides a summary of the distribution and dimensions of rooms. Moreover, the aggregated data is finally combined with additional features such as Hotel_Rating, Facility, and Price_RP through the merge function. For instance, 'Hotel C' is a two-star facility that includes 19 standard rooms with an average size of 31.5 square meters and 4 suite rooms with an average size of 36 square meters. The normalized dataset is displayed in Table 4.

Table 3. Data before normalize

Name	Rooms_Standard	Avg_Size_Standard	Rooms_Suite	Avg_Size_Suite	Hotel_Rating	Price_RP	Room_Type
A	14	18	0	0	Non Stars	200000	Standard
A	26	18	0	0	Non Stars	150000	Ekonomi
B	12	14	0	0	Stars 1	200000	Deluxe
B	12	14	0	0	Stars 1	250000	Deluxe
B	12		0	0	Stars 1	75000	standard
C			4	36	Stars 2	350000	VIP Room
C	2	30			Stars 2	300000	Double Twin
C	7	30			Stars 2	210000	Twin Bed
C	4	30			Stars 2	200000	Standard
C	6	36			Stars 2	180000	Family

Table 4. Data after normalize

Name	Rooms_Standard	Avg_Size_Standard	Rooms_Suite	Avg_Size_Suite	Hotel_Rating	Price_RP
A	40	18	0	0	0	200000
A	40	18	0	0	0	150000
B	36	14.67	0	0	1	200000
B	36	14.67	0	0	1	250000
B	36	14.67	0	0	1	75000
C	19	31.5	4	36	2	350000
C	19	31.5	4	36	2	300000
C	19	31.5	4	36	2	210000
C	19	31.5	4	36	2	200000
C	19	31.5	4	36	2	180000

Testing scenario

RF and CART classifiers predict hotel star ratings across different test sizes and random state values to evaluate performance. The dataset is split into training and testing subsets, employing test sizes of 20%, 30%, and 40%, along with random state values of 0, 42, and 100. This technique ensures the model across different data splits. The dataset was balanced using SMOTE to fix the class imbalance and ensure equal representation across all hotel star rating categories. For each combination of test size and random state, RF and CART classifiers are trained and optimized using GridSearchCV to assess different hyperparameter combinations and identify the optimal set systematically. The comparative analysis reveals several important findings about the test size and random state employed in the study. In every testing situation, RF consistently attained higher performance metrics in accuracy, precision, recall, and F1-score.

Comparison algorithm

This section analyzes the results of both algorithms in terms of test size and the random state, as evidenced by Table 5 and Figure 4. The findings indicate that RF consistently outperformed CART in both accuracy and F1-score across all test sizes and random states. The classifier achieved a maximum accuracy of 0.9932 and an F1-score of 0.9932, utilizing a random state of 100 and a test size of 0.4. The maximum value for CART was determined to be 0.9864 at both 100 and 0.2.

Table 5. Performance metric

Random State	Test Size	RF Accuracy	RF F1-Score	CART Accuracy	CART F1-Score
0	0.2	0.9636	0.9639	0.9409	0.9414
0	0.3	0.9758	0.9759	0.9667	0.9668
0	0.4	0.9682	0.9682	0.9591	0.9592
42	0.2	0.9909	0.9909	0.9818	0.9818
42	0.3	0.9788	0.9789	0.9758	0.9760
42	0.4	0.9773	0.9774	0.9659	0.9661
100	0.2	0.9909	0.9909	0.9864	0.9864
100	0.3	0.9909	0.9909	0.9788	0.9789
100	0.4	0.9932	0.9932	0.9591	0.9592

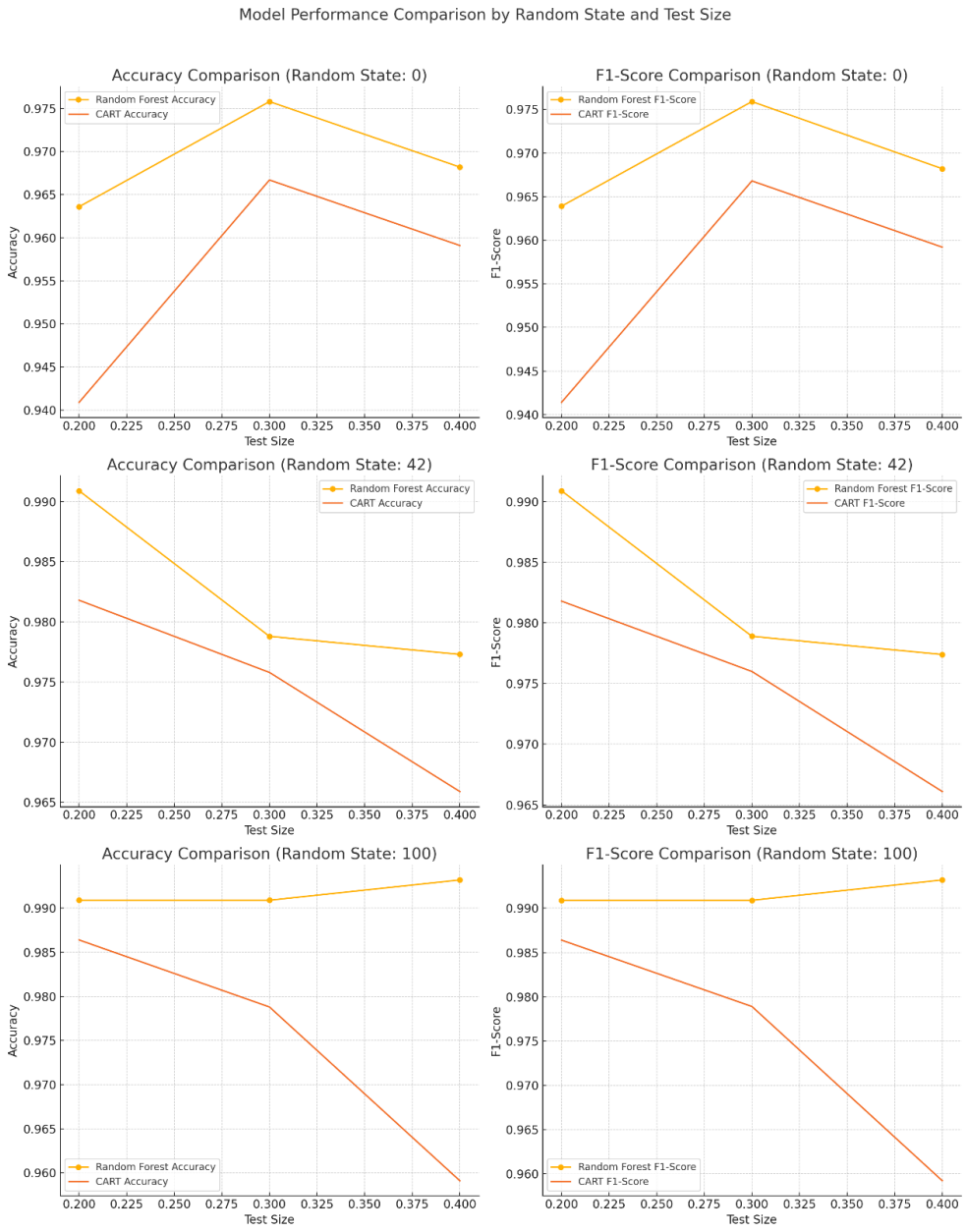

Figure 4. Model performance comparison

The detailed explanation presented in Table 4 shows that RF achieves an accuracy range between 0.9636 and 0.9758 at random state 0, with the highest recorded at a test size of 0.3, while the corresponding F1-score was from 0.9639 to 0.9759. Besides that, the results showed that the accuracy of CART for the same random state ranged from 0.9409 to 0.9667, with the highest observed at a test size of 0.3 and F1-score ranged from 0.9414 to 0.9668. For random state 42, RF shows even better results with accuracies ranging from 0.9773 to 0.9909 and F1-score from 0.9774 to 0.9909, indicating robust performance across different test sizes. Meanwhile, CART had from 0.9659 to 0.9818 and 0.9661 to 0.9818, respectively, with the best performance also recorded at the smallest test size of 0.2. Last but not least, for random state 100, RF reached peak performance with accuracies ranging from 0.9909 to 0.9932 and F1-score from 0.9909 to 0.9932. The scores for CART varied between 0.9591 and 0.9864, as well as 0.9592 to 0.9864, respectively.

The consistent outperformance of RV over CART was evident from the detailed metrics. The high and stable performance at different variations of random state and test size suggested the robustness and suitability of RV for classification tasks compared to CART. Moreover, a prediction process was used to compare the performance of each algorithm based on the level of accuracy, precision, and recall. Moreover,

the results were validated through the actual data obtained on the classification of star hotels in Banyumas Regency to determine the statistical significance of the differences.

In the class 0 performance graph presented in Figure 5, the most consistent performance was found at random state 42 and test size of 0.2 with RF reported to have 0.9909 accuracy, 0.9787 precision, 1.0 recall, and 0.9892 F1-score. CART also had 0.9818, 1.0, 1.0, and 1.0, respectively.

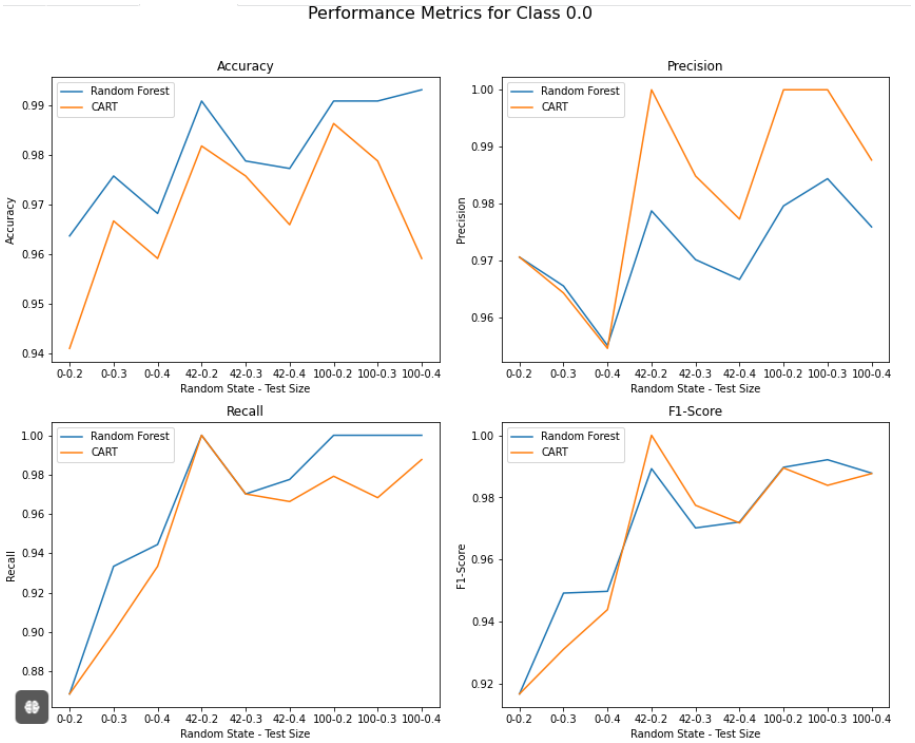

Figure 5. Performance metric for class 0.0

In the class 1 performance graph presented in Figure 6, the most consistent level of performance was found in random state 42 with test size of 0.2 as indicated by 0.9909 accuracy, 0.9787 precision, 1.0 recall, and 0.9892 F1-score recorded for RF. Meanwhile, the values for CART were recorded to be 0.9818, 0.9565, 0.9565, and 0.9565 respectively.

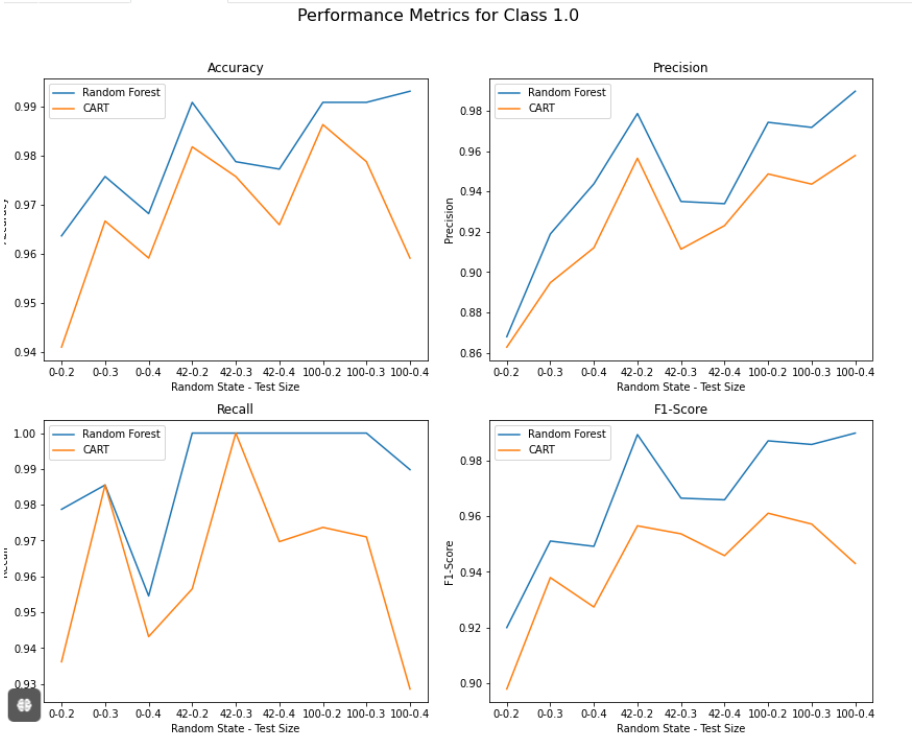

Figure 6. Performance metric for class 1.0

In the class 2 performance graph presented in Figure 7, the most consistent level was also at random state 42 with test size of 0.2 as reported by 0.9909 accuracy, 1.0 precision, 1.0 recall, and 1.0 F1-score for RF. Meanwhile, the values for CART were 0.9818, 0.9756, 1.0, and 0.9876, respectively.

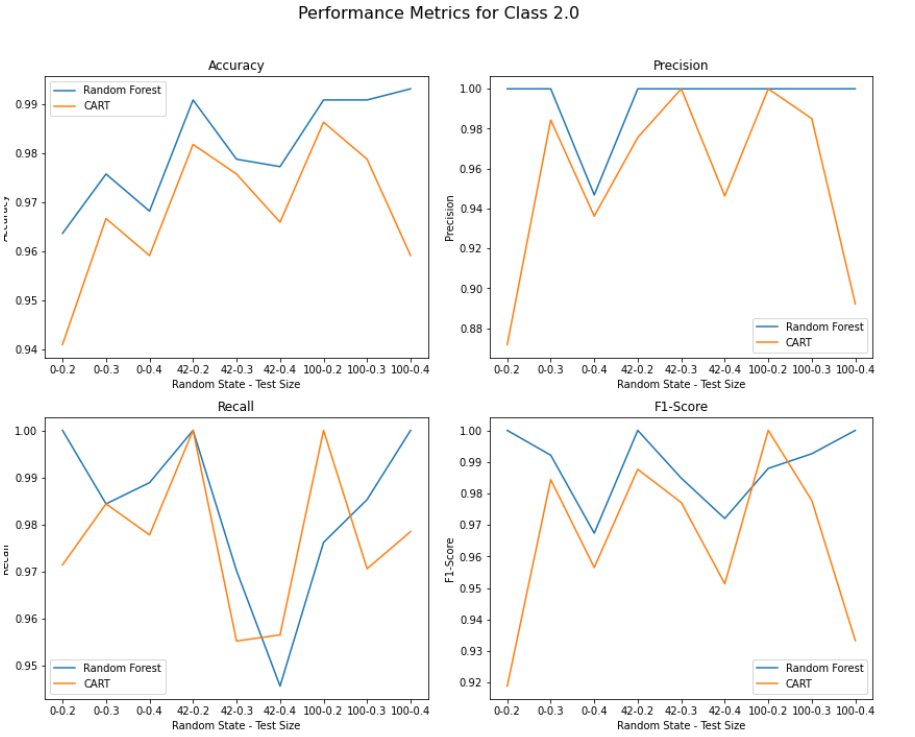

Figure 7. Performance metric for class 2.0

In the class 3 performance graph presented in Figure 9, the most consistent performance was in random state 42 with test size of 0.3 as reported in 0.9788 accuracy, 1.0 precision, 0.9552 recall, and 0.9771 F1-score for RF. Meanwhile, the respective values for CART were 0.9758, 1.0, 0.9552, and 0.9771 respectively.

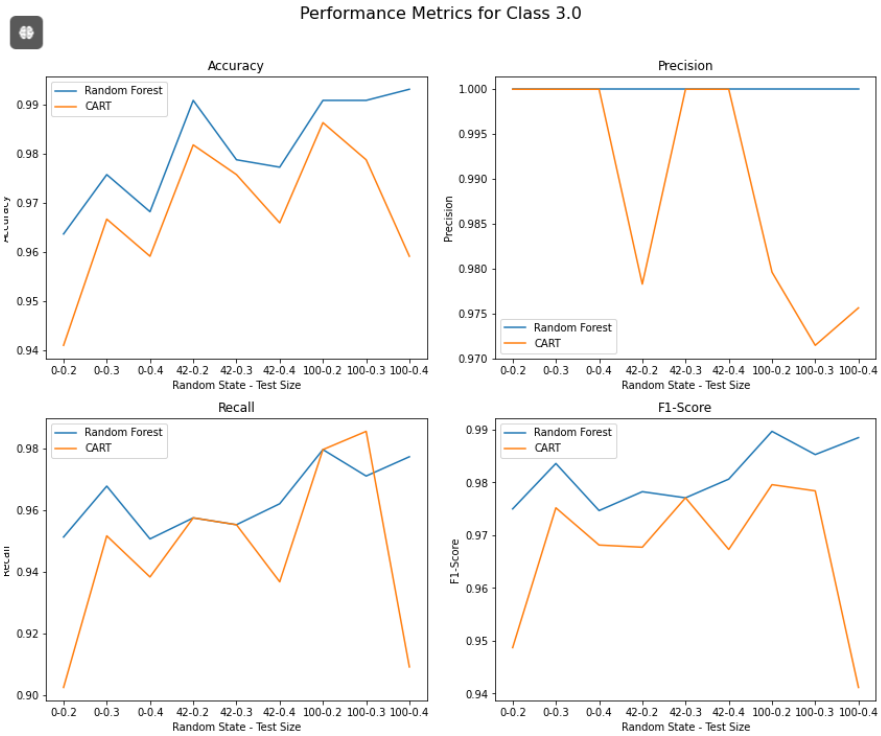

Figure 8. Performance metric for class 3.0

In the class 4 performance graph presented in Figure 9, the most consistent performance was found at random state 42 with test size of 0.2, where RF had 0.9909 accuracy, 1.0 precision, 1.0 recall, and 1.0 F1-score. Meanwhile, CART had 0.9818, 1.0, 1.0, and 1.0, respectively, with the same variation.

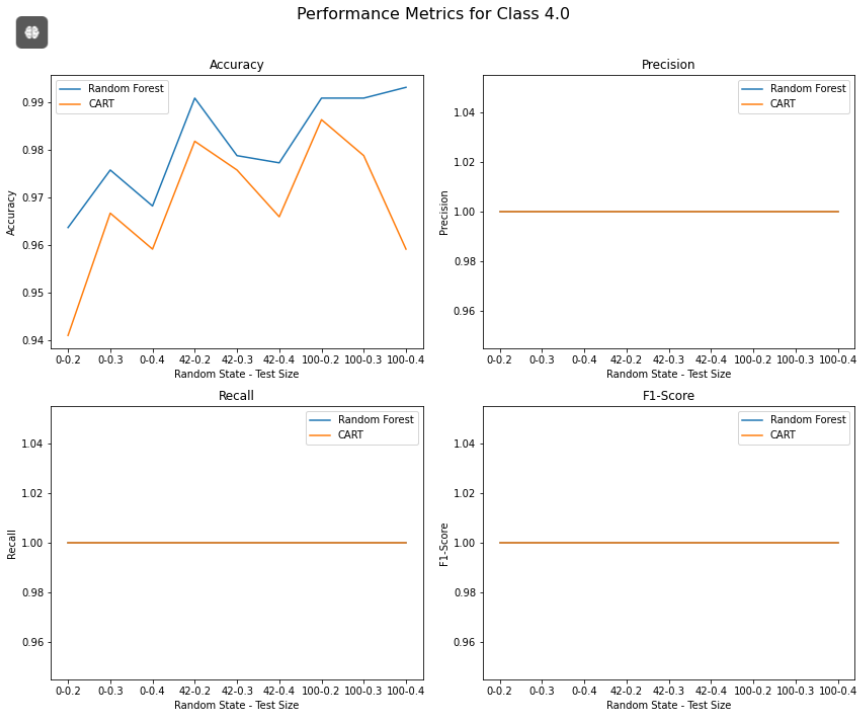

Figure 9. Performance metric for class 4.0

The graphs in Figure 10 compare RF and CART classifiers across different random states of 0, 42, and 100, as well as test sizes of 0.2, 0.3, and 0.4 based on accuracy and F1-score.

Accuracy Comparison:

- Random Forest consistently outperforms CART in terms of accuracy for all random states and test sizes.
- The accuracy of Random Forest ranges from 0.9636 to 0.9932, with the highest accuracy observed at Random State 100 and a test size of 0.4.
- CART's accuracy ranges from 0.9409 to 0.9864, with the best performance at Random State 100 and a test size of 0.2.

F1-Score Comparison:

- Random Forest also maintains a superior F1-Score across all conditions, indicating its robustness and reliability in classification tasks.
- The F1-scores for Random Forest range from 0.9639 to 0.9932, peaking at Random State 100 and a test size of 0.4.
- CART's F1-scores vary from 0.9414 to 0.9864, with the highest score at Random State 100 and a test size of 0.2.

The graph indicates that RF consistently outperforms CART in performance parameters, regardless of random state or test size. This indicates that RF was a more efficient and reliable classifier for the given dataset.

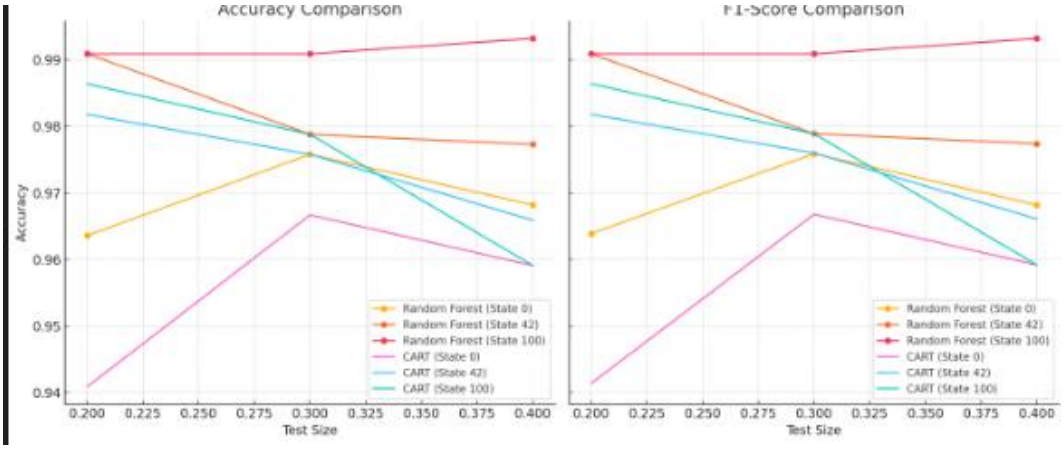

Figure 10. The Comparison between random forest and CART classifier

CONCLUSION

The results of this study indicate that RF generally outperformed CART in terms of both accuracy and F1-score across all random states and test sizes. The maximum accuracy with a value of 0.9932 and F1 score of 0.9932 achieved by RF occurred at a random state of 100 and a test size of 0.4, whereas CART recorded a value of 0.9864 for both metrics at 100 and 0.2, respectively. In the case of random state 0, the RF accuracy range was 0.9636 to 0.9758, with the maximum value observed at a test size of 0.3. The corresponding F1-score was 0.9639 to 0.9759. Simultaneously, CART showed values ranging from 0.9409 to 0.9667, with specific values of 0.9414 and 0.9668, respectively. RF accuracy for the random state 42 condition ranged from 0.9773 to 0.9909, with F1-score of 0.9774 to 0.9909. CART accuracy ranged from 0.9659 to 0.9818 and 0.9661 to 0.9818. For 100 random state conditions, the accuracy of RF varied from 0.9909 to 0.9932, with F1-score in the same range, whereas CART showed an accuracy from 0.9591 to 0.9864 and an F1-score from 0.9592 to 0.9864. The consistent superiority of RF over CART across various test sizes and random states illustrates the robustness and suitability for classification tasks. This analysis highlights the importance of thoroughly exhaustively validating classification models against actual data to evaluate the statistical significance of performance gaps between algorithms and certain accuracy. Specifically, the RF model with a random state of 42 and a test size of 0.2 showed the most consistent performance regarding accuracy, precision, recall, and F1-score.

REFERENCES

[1] R. Yusuf and M. Veranita, "Minat Berwisata Kaum Milenial Di Era New Normal," *J. Kepariwisataan Indones. J. Penelit. dan Pengemb. Kepariwisataan Indones.*, vol. 15, no. 2, pp. 158–167, 2021, doi: 10.47608/jki.v15i22021.158-167.

[2] S. Wijayanto, D. A. Prabowo, D. Y. Kristiyanto, and M. Y. Fathoni, "Analisis Sentimen Berbasis Aspek pada Layanan Hotel di Wilayah Kabupaten Banyumas dengan Word2Vec dan Random Forest," *J. Inform. J. Pengemb. IT*, vol. 8, no. 1, pp. 1–3, 2022, doi: 10.30591/jpit.v8i1.4186.

[3] A. Utami, M. L. L. Usman, I. F. Ramadhani, S. N. F. Syam, and F. A. Fauzan, "Hotel Selection Decision Support System with the Simple Additive Weighting (SAW) Method," *Build. Informatics, Technol. Sci.*, vol. 4, no. 3, pp. 1181–1187, 2022, doi: 10.47065/bits.v4i3.2262.

[4] B. B. Baskoro, I. Susanto, and S. Khomsah, "Analisis Sentimen Pelanggan Hotel di Purwokerto Menggunakan Metode Random Forest dan TF-IDF (Studi Kasus: Ulasan Pelanggan Pada Situs TRIPADVISOR)," *INISTA (Journal Informatics Inf. Syst. Softw. Eng. Appl.*, vol. 3, no. 2, pp. 21–29, 2021, doi: 10.20895/INISTA.V3.

[5] T. Sufi and S. P. Singh, "Hotel classification systems: A case study," *Prabandhan Indian J. Manag.*, vol. 11, no. 1, pp. 52–64, 2018, doi: 10.17010/pijom/2018/v11i1/120823.

[6] I. W. P. Pratama and O. Asroni, "Analysis of hotel ratings and price range in labuan bajo, Indonesia," *J. Mantik*, vol. 6, no. 4, 2023.

[7] I. Z. P. Hamdan, M. Othman, Y. M. M. Hassim, S. Marjudi, and M. M. Yusof, "Customer Loyalty Prediction for Hotel Industry Using Machine Learning Approach," *Int. J. Informatics Vis.*, vol. 7, no. 3, pp. 695–703, 2023, doi: 10.30630/joiv.7.3.1335.

[8] A. Vagena and P. La, "Characteristics of official hotel classification systems," *Int. J. Res. Tour. Hosp.*, vol. 6, no. 3, pp. 33–49, 2020, doi: 10.20431/2455-0043.0603004.

[9] N. Čumlievski, M. Brkić Bakarić, and M. Matetić, "A Smart Tourism Case Study: Classification of Accommodation Using Machine Learning Models Based on Accommodation Characteristics and Online Guest Reviews," *Electron.*, vol. 11, no. 6, 2022, doi: 10.3390/electronics11060913.
[10] A. Subroto and M. Christianis, "Rating prediction of peer-to-peer accommodation through attributes and topics from customer review," *J. Big Data*, vol. 8, no. 1, 2021, doi: 10.1186/s40537-020-00395-6.
[11] K. Kozlovskis, Y. Liu, N. Lace, and Y. Meng, "Application of Machine Learning Algorithms To Predict Hotel Occupancy," *J. Bus. Econ. Manag.*, vol. 24, no. 3, pp. 594–613, 2023, doi: 10.3846/jbem.2023.19775.
[12] Y. Azhar, G. A. Mahesa, and M. C. Mustaqim, "Prediction of hotel bookings cancellation using hyperparameter optimization on Random Forest algorithm," *J. Teknol. dan Sist. Komput.*, vol. 9, no. 1, pp. 15–21, 2021, doi: 10.14710/jtsiskom.2020.13790.
[13] M. A. Afrianto and M. Wasesa, "Booking Prediction Models for Peer-to-peer Accommodation Listings using Logistics Regression, Decision Tree, K-Nearest Neighbor, and Random Forest Classifiers," *J. Inf. Syst. Eng. Bus. Intell.*, vol. 6, no. 2, p. 123, 2020, doi: 10.20473/jisebi.6.2.123-132.
[14] A. Khairunnisa, K. A. Notodiputro, and B. Sartono, "A Comparative Study of Random Forest and Double Random Forest Models from View Points of Their Interpretability," *Sci. J. Informatics*, vol. 11, no. 1, pp. 207–218, 2024, doi: 10.15294/sji.v11i1.48721.
[15] 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, 2021, doi: 10.3390/a14110318.
[16] Riska Chairunisa, Adiwijaya, and Widi Astuti, "Perbandingan CART dan Random Forest untuk Deteksi Kanker berbasis Klasifikasi Data Microarray," *J. RESTI (Rekayasa Sist. dan Teknol. Informasi)*, vol. 4, no. 5, pp. 805–812, 2020, doi: 10.29207/resti.v4i5.2083.
[17] F. Al Farikhi and R. W. D. Pramono, "Perbandingan algoritma Classification and Regression Tree (CART) dan Random Forest (RF) untuk klasifikasi penggunaan lahan pada google earth engine," *Spat. Wahana Komun. dan Inf. Geogr.*, vol. 23, no. 2, pp. 170–179, 2023, doi: 10.21009/spatial.232.09.
[18] H. L. Purwanto and J. W. Kuswinardi, "Pemilihan Hotel Menggunakan 'Technique for Order Preference By Similarity To Ideal Solution' Berbasis Webgis," *Kurawal - J. Teknol. Inf. dan Ind.*, vol. 3, no. 1, pp. 28–39, 2020, doi: 10.33479/kurawal.v3i1.302.
[19] L. D. Utami, "Komparasi Algoritma Klasifikasi Pada Analisis Review Hotel," *J. Pilar Nusa Mandiri*, vol. 14, no. 2, p. 261, 2018, doi: 10.33480/pilar.v14i2.1023.
[20] D. Hartanti, A. Ichsan Pradana, and S. Lestari, "Komprasi Algoritma Decision Tree, SVM dan ANN untuk Reservasi Hotel," *DutaCom*, vol. 16, no. 1, pp. 21–27, 2023, doi: 10.47701/dutacom.v16i1.2647.
[21] A. Budiyantara, A. K. Wijaya, A. Gunawan, and M. Rolland, "Analisis Data Mining Hotel Booking Menggunakan Model Id3," *JBASE - J. Bus. Audit Inf. Syst.*, vol. 4, no. 1, pp. 1–12, 2021, doi: 10.30813/jbase.v4i1.2728.
[22] M. Adil, M. F. Ansari, A. Alahmadi, J. Z. Wu, and R. K. Chakrabortty, "Solving the problem of class imbalance in the prediction of hotel cancelations: A hybridized machine learning approach," *Processes*, vol. 9, no. 10, 2021, doi: 10.3390/pr9101713.
[23] R. Timofeev, "Classification and Regression Trees theory and applications," *M.A. Berlin Humboldt Univ. Berlin.*, 2004, doi: 10.1016/B978-008045405-4.00149-X.
[24] J. Prasetya, S. I. Fallo, and M. A. Aprihartha, "Stacking Machine Learning Model for Predict Hotel Booking Cancellations Model Machine Learning Stacking untuk Prediksi Pembatalan Pemesanan Hotel," *J. Mat. Stat. dan Komputasi*, vol. 20, no. 3, pp. 525–537, 2024, doi: 10.20956/j.v20i3.32619.
[25] L. Sari, A. Romadloni, R. Lityaningrum, and H. D. Hastuti, "Implementation of LightGBM and Random Forest in Potential Customer Classification," *TIERS Inf. Technol. J.*, vol. 4, no. 1, pp. 43–55, 2023, doi: 10.38043/tiers.v4i1.4355.
[26] S. Mandala, T. Cai Di, M. S. Sunar, and Adiwijaya, "ECG-based prediction algorithm for imminent malignant ventricular arrhythmias using decision tree," *PLoS One*, vol. 15, no. 5, pp. 1–20, 2020, doi: 10.1371/journal.pone.0231635.
[27] I. Mabarti, "Implementation of Minimum Redundancy Maximum Relevance (MRMR) and Genetic Algorithm (GA) for Microarray Data Classification with C4. 5 Decision Tree," *J. Data Sci. Its ...*, no. January, pp. 38–47, 2020, doi: 10.34818/JDSA.2020.3.37.

[28] R. Pratiwi, M. N. Hayati, and S. Prangga, "Perbandingan Klasifikasi Algoritma C5.0 Dengan Classification and Regression Tree (Studi Kasus : Data Sosial Kepala Keluarga Masyarakat Desa Teluk Baru Kecamatan Muara Ancalong Tahun 2019)," *BAREKENG J. Ilmu Mat. dan Terap.*, vol. 14, no. 2, pp. 273–284, 2020, doi: 10.30598/barekengvol14iss2pp273-284.
[29] S. Aldiansyah and R. A. Saputra, "Comparison of Machine Learning Algorithms for Land Use and Land Cover Analysis Using Google Earth Engine (Case Study: Wanggu Watershed)," *Int. J. Remote Sens. Earth Sci.*, vol. 19, no. 2, p. 197, 2023, doi: 10.30536/j.ijreses.2022.v19.a3803.