Comparative analysis of classification algorithm: Random Forest, SPAARC, and MLP for airlines customer satisfaction

Safira Amalia[1], Irene Deborah[1], Intan Nurma Yulita[2,3]
[1]Master of Science Management, Universitas Padjadjaran, Indonesia
[2]Research Center for Artificial Intelligence and Big Data, Universitas Padjadjaran, Indonesia
[3]Department of Computer Science, Universitas Padjadjaran, Indonesia

Abstract
The airline business is one of the businesses determined by the quality of its services. Every airline creates its best service so that customers feel satisfied and loyal to using their services. Therefore, customer satisfaction is an essential metric to measure features and services provided. By having a database on customer satisfaction, the company can utilize the data for machine learning modelling. The model generated can predict customer satisfaction by looking at the existing feature criteria and becoming a decision support system for management. This article compares machine learning between Split Point and Attribute Reduced Classifier (SPAARC), Multilayer Perceptron (MLP), and Random Fores (RF) in predicting customer satisfaction. Based on the data testing, the Random Forest algorithm provides better results with the lowest training time compared to SPAARC and MLP. It has an accuracy of 95.827%, an F-score of 0.958, and a training time of 84.53 seconds.

INTRODUCTION

Every company competes to provide the best service and features to create and increase customer satisfaction in business activities. One of the businesses determined by the quality of its service is the airline business. Every airline provides its best service so consumers are satisfied and loyal to use the services of an airline so that the company can continue to grow and be able to compete in the industry. Passengers prefer to rate airlines based on their satisfaction with in-flight services [1]. So increasing the quality of in-flight service becomes one of the success factors of an airline. Evaluating the quality of services can be done by checking customer satisfaction. Customer satisfaction is an essential metric to measure consumer loyalty and intention to use services/products again, increase positive ratings, and reduce costs for new customer acquisitions [2].

Companies can use customer satisfaction surveys to gain consumer ratings and evaluate the features and services. Data obtained can be used as training data for machine learning supervised learning. In addition, the available data can be used for supervised learning. The machine learning training process produces a model to predict customer satisfaction by looking at the existing feature criteria. Furthermore, the model generated by machine learning can be used as a decision support system that helps management plan future business strategies and strategies for retaining customers and new customer acquisitions.

Many studies have been conducted to analyze customer satisfaction with airlines. Kumar and Zymbler [3] analyzed tweets for improving customer experience by using Support Vector Machine (SVM), Artificial Neural Networks (ANN), and Convolutional Neural Network (CNN).

The study showed that CNN improved the performance of the classification model and provided better results than ANN and SVM. Gracia et al. [4] used an ensemble regression model to analyze the problem of predicting customer satisfaction. The results showed ensemble regression produced the best results. Hulliyah [5] researched predicting flight passengers using a classification algorithm: KKN, Logistics Regression, Gaussian NB, Decision Tree, and Random Forest (RF). This study concentrates on the Wi-Fi service experience, and the algorithm that provides the best result is RF, with 99.00% accuracy at a threshold of 0.7.

With many classification algorithms in machine learning, this study chose to develop a classification using three models there are Split Point and Attribute Reduced Classifier (SPAARC), Multilayer Perceptron (MLP), and Random Forest (RF). SPAARC is a new method, and not many studies have used this algorithm, so this study aims to test the SPAARC algorithm and compare it with other algorithms in a case. The advantage of the SPAARC method is reducing the computational workload process from the decision tree by selecting attributes dynamically or using the tree depth levels involved [6]. Another algorithm used in this study is Random Forest, which gives high accuracy based on previous studies. On the other hand, the MLP algorithm is used due to its ability to classify large amounts of data with various features.

This paper contains a comparative analysis of several classification algorithms. Accuracy results from the data collection will be obtained, and this study can show which algorithm has a high and good level of accuracy according to the existing parameters.

MATERIAL AND METHOD
Supervised Learning
Supervised learning is defined where the class labels are known and the class limits are well represented in the data set [7][8]. Several methods that are included in supervised learning are Decision Trees, Naive Bayes, Neural Networks, and Deep Learning. The function of supervised learning is to build a classifier by providing a classified training data set [9]. This method processes the training data set to find the input and target attribute relationship. Finally, the assembled model will be used to predict the attribute target value for the new data set. The challenge of supervised learning is a generalization, where the classifier model has to be appropriately used on all data.

The paradigm of supervised learning can be seen from the Neural Network, which includes the MLP algorithm, which can efficiently find solutions for several linear and non-linear problems, such as the classification process [10][11]. Moreover, processes in MLP have unique characteristics, such as; 1) Nonlinearity that is reflected in the activity and can be distinguished, 2) One or more hidden layers of neurons to enable the network to solve complex problems, and 3) Interconnection model.

Meanwhile, the decision trees are used for prediction functions such as classification and regression. The nodes represent the data set features, and the branches represent the rules of the decisions [12]. This decision tree has two nodes: the decision node and the leaf node. The decision node is used to make any decision and has many branches, while the leaf node is the output of the decision and does not have branches [13].

Split Point and Attribute Reduced Classifier (SPAARC)
Split Point and Attribute Reduced Classifier (SPAARC) is one of the classification tree algorithms from the Decision Tree or Classification (CART) and Regression Tree method [6]. SPAARC has two components in dealing with decision tree problems. This technique is used to reduce the computational process and increase processing time while minimizing the accuracy of the classification process. The SPAARC method is applied to the classification algorithm by implementing split-point numerical attribute analysis and recursive selection of attribute nodes. The process includes split-point sampling to reduce the number of these split points when used in testing the suitability of attributes at each node in the decision tree and usage of node-attribute sampling to test each alternative horizontally at the tree node level.

Components of SPAARC consist of Node Attribute Sampling (NAS) and Split-Point Sampling (SPS). The purpose of the combined NAS components is to balance the different requirements of classification accuracy and processing time [6]. The research found supporting evidence on optimizing the speed of induction of decision trees studied by Fayyad and Irani [14] by using entropy as a heuristic in decision trees. Yates et al. [6] proposed the NAS component contributes to avoiding testing every non-class attribute in each tree node. It dynamically selects the attribute space by switching between complete attributes lists and

subsets—based on the depth of the tested nodes. At the same time, the SPS component can reduce the number of possible split points tested dynamically even though the SPS component only handles numeric attributes. These two components of the SPAARC algorithm can improve time savings during the modelling process by accelerating the pruning process. However, the improvement of SPAARC can eliminate classification accuracy, which contributes significantly to implementing the dataset.

The hyperparameters in SPAARC are minNumObj (M), numFoldPruning (N), size (C), and seed (S). minNumObj (M) is the minimum number of branches on a node. The smaller the minNumObj value, the less branching in a node takes a longer processing time than a larger minNumObj. The second is numFoldPruning (N) which is trimming the amount of data to reduce pruning errors in each tree. Pruning on the decision tree can reduce outliers and data noise to increase accuracy in data classification. The third is size per (C) is a percentage of the training data set size. Last is the seed (S), which sets a local random seed for randomization.

Multilayer Perceptron (MLP)

Multilayer Perceptron (MLP) is part of the Artificial Neural Network (ANN). The Neural Network method starts with receiving input and performs operations with a weight, adding them (weighted sum) and adding bias. This operation will be used as a parameter of the activation function, which will be the neuron's output. MLP is a neural network structure that is widely used and consists of 3 layers (layers) of structure, namely the input layer, output layer, and hidden

layer. Each layer contains several neurons (nodes) depending on how complex the process is. Neurons in MLP are trained with a backpropagation algorithm. [15] MLP is commonly used for classification, recognition, prediction, and forecasting activities.

MLP works by moving the data forward from the input layer toward the output layer as depicted in Figure 1. MLP works starting from the input layer receiving the input signal for processing. Then the input is processed by the MLP computing engine in the hidden layer, which is located between the input and output layers. Finally, tasks that need to be done, and the computing results, are carried out by the output layer.

In neural networks, hyperparameters determine the structure of the neural network and how the model is trained. The hyperparameter tuning process is the key to reducing the computation time that gives a reasonable error. Hyperparameters that can be adjusted in MLP are the number and size of hidden layers (the depth of the algorithm model), learning rate, momentum, and dropout rate [16].

The learning rate sets the minimum step for each iteration. Setting the learning rate can result in the model's speed to produce the model and solution (example: minimum error). A small learning rate can produce a smoother model and more minor errors than a significant learning rate. Then, momentum in the neural network is a weight change based on the direction of the gradient of the last pattern with the previous pattern. The use of the momentum parameter affects the learning process towards a faster and more stable convergence.
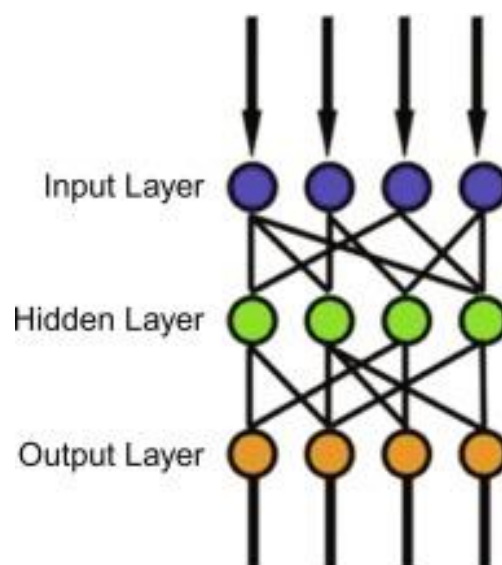

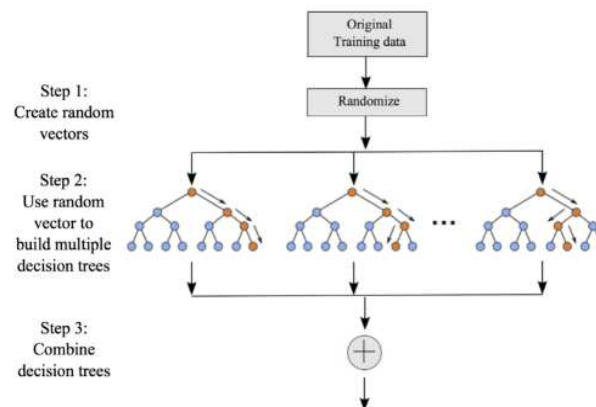Figure 1. Multilayer Perceptron Block Diagram [17]

Figure 2. Random Forest Block Diagram [17]

Random Forest

Random Forest is a machine learning algorithm with an ensemble method that can be used for classification and regression. The Random Forest consists of a collection of decision trees associated with bootstrap samples collected from the original dataset as shown in Figure 2.

The nodes are divided based on the entropy of the selected feature subset. S. Suthaharan [7] explained that the subset formed from the original dataset has the same size as the original dataset by bootstrapping. The advantage of using the Random Forest method compared to the decision tree is that it provides several classifications from several decision trees in the testing phase.

In addition, the accuracy of the Random Forest is higher. It retains some good qualities in the decision tree, such as interpreting the relationship between predictors and outcomes [18]. These characteristics make it a preferred method for a decision tree. Suthaharan [7] identified it as a good technique for solving classification problems in big data because of its flexible parallel structure that works with technologies to handle big data, such as Hadoop, MapReduce, etc.

The Random Forest algorithm has several hyperparameters that the researcher can set. By selecting the hyperparameters, the model can perform better. The hyperparameters used in Random Forest consist of the structure of each tree (minimum number of node sizes), forest structure and size (number of trees), and categorical elements (number of variables considered in each branch/try) [19].

The number of node sizes sets the minimum number of observations on the terminal nodes. Setting leads at low trees, and high depths produces more branches to reach the terminal nodes by setting the number of node sizes. The higher the result is on reducing computation time without reducing the prediction performance. At the same time, the number of trees is a parameter that is recommended to be set in large values. More trees result in good modelling.

Cross-Validation

Cross-validation or rotation estimation is a model validation technique used to assess the statistical results of the analysis to be generalized from the component data set [20]. Cross-validation can be used for estimating errors in predicting or evaluating the performance of the model [21]. In cross-validation, rotation estimation is known and divides the data into k subsets of almost the same size. Then training and testing are conducted as many as k; in each repetition, one set will be used for test data while the other k data subgroups serve as training data. K-fold is known for evaluating the classifier's performance, where the K-Fold method can be used if the amount of data is limited. The best implementation of the number of folds in the validity test uses 10-fold cross-validation in each model [22]. Cross-validation is also a validation method used to increase the accuracy of the algorithms of other methods used.

Method

The learning methods that are used in the comparative analysis are SPAARC, MLP, and Random Forest. Each method uses the stages of the Knowledge Discovery in Database (KDD). Data collection and selection, preprocessing/cleaning, transformation, data mining, and interpretation/evaluation as shown in Figure 3 [23].
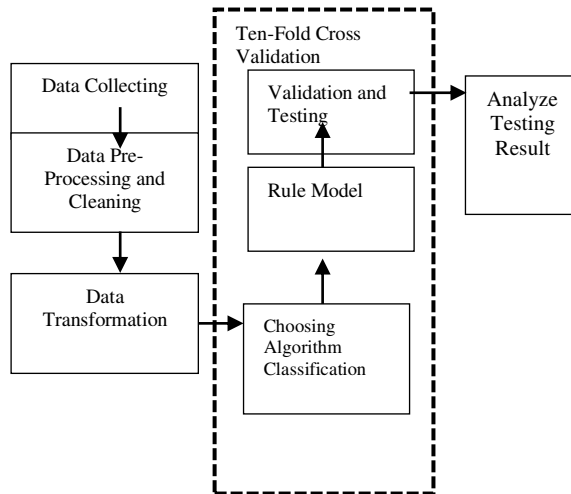
Figure 3. Research Method [24]

Table 1. Data Features

Features	Description
Gender	Customer's gender (male/female)
Consumer type	Type of customer (loyal/disloyal)
Age	Customer's age
Type of travel	Travel purpose (business/personal)
Class	Type of class (Eco/Business/Eco Plus)
Flight distance	Flight distance
Seat comfort	Rating of seat comfort (Likert scale 1-5)
Departure/Arrival time convenient	Rating of departure/arrival time convenient (Likert scale 1-5)
Food & drink	Rating of food and drink (Likert scale 1-5)
Gate location	Rating satisfaction of gate location (Likert scale 1-5)
Inflight Wi-Fi service	Rating satisfaction of Wi-Fi services (Likert scale 1-5)
Inflight entertainment	Rating of inflight entertainment (Likert scale 1-5)
Online support	Rating satisfaction of online support (Likert scale 1-5)
Ease of Online booking	Rating satisfaction of online booking feature (Likert scale 1-5)
Onboard service	Rating satisfaction of onboard services (Likert scale 1-5)
Legroom service	Rating satisfaction of legroom service (Likert scale 1-5)
Baggage handling	Rating satisfaction of baggage handling (Likert scale 1-5)
Checkin service	Rating satisfaction of check-in service (Likert scale 1-5)
Online boarding	Rating satisfaction of baggage handling (Likert scale 1-5)
Cleanliness	Rating satisfaction of airplane's cleanliness (Likert scale 1-5)
Departure Delay in Minutes	Departure delay duration (in minutes)
Arrival Delay in Minutes	Arrival delay duration (in minutes)

The process starts with data collection, where the data is obtained from Kaggle. The dataset consists of 129,880 data entries, 22 features, and 2 class labels (satisfied and dissatisfied). The detail of data features is as listed in Table 1. The next step is to do data pre-processing and cleaning, where the pre-processing stage aims to increase the data's quality before processing. In this phase, the data are cleaned to fit on a Likert scale of 1-5. A score of 0 is considered an unanswered survey statement.

After deleting data with a score of 0 and statements that are not filled in, the total data that can be processed for modelling is 119,611 data entries. After the data set is ready, algorithms are chosen and used for rule models for research. Next, validation and testing were carried out to determine the prediction results' accuracy, precision, recall, and classification error to find out the data results. Finally, an analysis of the test results is carried out by the discussion and then will be compared between the classification algorithms that have been determined. After data preprocessing, data is ready to be used for machine learning with the selected method. Transformation data is involved when needed.

Then, for model testing, there is a validity test to measure the level of accuracy, f1score value, classification error, and training time. Then, last is analyzing the results and comparing the modelled algorithm.

Modelling

The learning method used in this study is SPAARC, MLP, and RF SPAARC, a new algorithm, while MLP and Random Forest are the most popular algorithms often used in large data sets.

There are hyperparameter settings in each algorithm model used (SPAARC, MLP, and Random Forest) to improve the algorithm's performance.

The modelling uses a 10-fold cross-validation technique. The device used for modelling is a 1.6 GHz Dual-Core Intel Core i5 with OS version Big Sur Version 11.3.1, 4 GB 1600 MHz DDR3. Every hyperparameter in the method is listed in Table 2.

After applying the algorithm method, the performance is measured by several metrics which are then used for comparative analysis. This paper uses the metrics commonly used in classification, accuracy and F-score. Accuracy measures how much the model can classify the data correctly. Calculations do not discriminate between the correct number of labels from

different classes [25]. Accuracy can be calculated by using (1).

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN}$$ (1)

where:
TP = true positive
TN = true negative
FP = false positive
FN = false negative

Table 2. Method and Hyperparameter Used

Method	Hyperparameter
SPAARC	minNumobj, numFoldPruning, percentage of training data, number of seeds
MLP	Learning rate, hidden layer, momentum
RF.	Number of the decision tree, node size

Meanwhile, F-score is a calculation with weighting from precision (the accuracy of the model to predict positive labels) and recall (how much actual positive data can be captured by the model with positive data labels (true positive)). F-score aims to measure the effectiveness of the method used. F-score can be calculated by using (2).

$$F1 = 2 \times \frac{Precision \times Recall}{Precision + Recall}$$ (2)

RESULTS AND DISCUSSION
SPAARC Model Testing

The hyperparameter was tested four times to optimize the SPAARC model. As for the first test by changing minNumobj (M), the M values used in the experiment are 2.0, 1.0, and 0.75. The test results are listed in Table 3.

The most optimal accuracy and F-score results are obtained at the M value of 1.0 with an accuracy rate of 95.05%. The smaller the value of M causes the model training time to be longer with the same accuracy results. The second hyperparameter test is numFoldPruning (N), where the N values used in the experiment are 2,3 and 5. The test results are shown in Table 4.

NumFoldPruning (N), which sets pruning the amount of data to reduce pruning errors on each tree, produces the highest accuracy when the number of pruning is 5 with a previously determined M value of 1.0 and an accuracy value of 95.054%.

The third Hyperparameter test is to test the training data set (C) percentage, where the C values used in this experiment are 1, 0.75, and 0.5. So that the test results are obtained as follows in Table 5. The last hyperparameter test is S. The results are shown in Table 6.

Table 3. Accuracy & F-Score Result of MinNumObj (M) Setting

M	N	C	S	F-score	Accuracy (%)	Training Time (s)
2.00	5	1.0	1	0.950	95.041	76.290
1.00	5	1.0	1	0.951	95.054	84.610
0.75	5	1.0	1	0.951	95.054	94.810

Table 4. Accuracy & F-Score Result of NumFoldPrunning (N) Setting

M	N	C	S	F-score	Accuracy (%)	Training Time (s)
1.0	5	1.0	1	0.951	95.054	84.610
1.0	3	1.0	1	0.950	94.999	55.560
1.0	2	1.0	1	0.949	94.895	46.730

Table 5. Accuracy & F-Score Result of Training Data (C) Setting

M	N	C	S	F-score	Accuracy (%)	Training Time (s)
1.0	5	1.00	1	0.951	95.054	84.610
1.0	5	0.75	1	0.949	94.921	61.950
1.0	5	0.50	1	0.948	94.789	41.950

Table 6. Accuracy & F-Score Result of Number of Seeds (S) Setting

M	N	C	S	F-score	Accuracy (%)	Training Time (s)
1.0	5	1.0	1	0.951	95.054	84.610
1.0	3	1.0	5	0.951	95.071	86.550
1.0	2	1.0	9	0.951	95.067	88.980

The best accuracy is 95.07%. It was obtained when the number of seeds S= 5. From the test results with hyperparameter settings, the SPAARC algorithm produces the best model with values of M 1, N 5, C 1, and S 5. The model has an accuracy rate of 95.07%. The F-score value is 0.951, and a training time of 86.55 seconds.

Multilayer Perceptron Model Testing

The first hyperparameter test is the Learning rate which is used for model optimization in MLP. The learning rates used in the experiment were 0.3, 0.01, and 0.001. The test results are listed in Table 7.

The highest accuracy occurs in the 0.01 learning rate setting, with 94.78% in the learning rate test. The test results also found that the higher the learning rate, the shorter the training time. Next is testing the number of hidden layer hyperparameters. The MLP model used is the best learning rate setting in the previous test, 0.01. The number of hidden layers tested was 5, 10, and 15. Table 8 lists the results of the hyperparameters.

It increases the number of hidden layers resulting in a higher level of accuracy. This can be seen in the most significant number of hidden layers, 15, which has an accuracy rate of 94.85% and the F-score is 0.948. The higher the hidden layer value also affects the longer training time.

The last test is momentum using hidden layers and learning rates that produce the highest accuracy. The amount of momentum tested is 0.2, 0.5, and 0.75. Table 9 shows the results of the hyperparameter testing.

Changes in the momentum value at a certain point can cause an increase in accuracy and F1score. For example, this experiment with a momentum value of 0.5 produces an accuracy of 94.94% and an F-score of 0.949. The test results with hyperparameter settings show that the MLP algorithm makes the best model with a learning rate of 0.01, hidden layers 15, and a momentum

value of 0.5. The model has an accuracy of 94.94% and an F-score value of 0.949.

Table 7. Accuracy & F-Score Result of Learning Rate Setting

Learning Rate	F-Score	Accuracy (%)	Training Time (s)
0.300	0.945	94.497	324.420
0.010	0.948	94.777	331.420
0.001	0.944	94.367	336.980

Table 8. Accuracy & F-Score Result of Number of Hidden Layer Setting

Num of Hidden Layer	Learning Rate	F-Score	Accuracy (%)	Training Time (s)
5	0.01	0.933	93.275	152.060
10	0.01	0.947	94.650	272.290
15	0.01	0.948	94.846	391.890

Table 9. Accuracy & F-Score Result of Momentum Setting

Momentum	Hidden layer	Learning Rate	F-score	Accuracy (%)	Training Time (s)
0.2	15	0.01	0.948	94.846	391.890
0.5	15	0.01	0.949	94.941	394.260
0.75	15	0.01	0.949	94.863	412.230

Random Forest Model Testing

In the Random Forest algorithm, the hyperparameters that need to be set are the number of decision trees and node sizes to improve the performance. Therefore, the first hyperparameter setting is to test changes in the number of decision trees. The number of decision trees tested in the experiment was 100, 80, and 50, as listed in Table 10.

In setting the number of decision trees, there is an increase in accuracy and F-score if the number of decision trees is increased. For example, the highest accuracy is obtained from the hyperparameter setting with a total decision tree of 100 with a 95.8% accuracy rate and an F-score value of 0.958. The result is similar to Probst's [19] findings which are the training time increases linearly with the number of trees. Next is setting the maximum number of node sizes listed in Table 11.

Table 10. Accuracy & F-Score Result of Number Decision Tree Setting

Num of Decision Tree	Accuracy (%)	F-score	Training Time (s)
100	95.827	0.958	82.800
80	95.799	0.958	70.840
50	95.744	0.957	41.560

Table 11. Accuracy & F-Score Result of Max Node Size Setting

Max node size	Num of Decision Tree	Accuracy (%)	F-score	Training Time (s)
20	100	95.765	0.958	86.870
50	100	95.827	0.958	86.330
100	100	95.827	0.958	84.530

The model's accuracy is the same as setting the maximum number of node sizes to 50 and the maximum number of node sizes to 100. The higher the maximum number of node sizes, the shorter the training time. From the test results with hyperparameter settings, the Random Forest algorithm produces the best model with a decision tree number of 100 and a maximum number of node size 100. The model has a 95.827% accuracy rate, and the F-score value is 0.958.

Discussion

After calculating accuracy, F-score, and training time with the specified airline dataset and comparing SPAARC, MLP, and RF produced like the data in Table 12. In each of these comparison algorithms, using the 10-validity test fold cross-validation in each model, the accuracy results and the highest F-score are obtained using the Random Forest algorithm. It has the highest accuracy and score and the lowest training time compared to the other two algorithms, where RF has a 95.837% accuracy rate, F-score is 0.958, and a training time of 84.53 seconds.

Table 12. Result Comparison

Metode	Accuracy (%)	F-score	Training Time (s)
SPAARC (M 1, N 5, C 1, S 5)	95.071	0.951	86.550
Multilayer Perceptron (momentum 0.5, hidden layer 15, learning rate 0.01)	94.941	0.949	394.260
Random Forest (Num of decision tree 100, max node size 100)	95.827	0.958	84.530

Meanwhile, SPAARC has a 95.071% accuracy rate, an F-score of 0.951, and a training time of 86.55. Then, MLP has a 94.941% accuracy rate, an F-score of 0.949, and a training time of 394.26 seconds. In a previous study, SPAARC had minimal effect on decision tree classification accuracy and reduced training time by 70% [6]. In this study, the accuracy of SPAARC reached 95% and had a training time of 86 seconds. The results of SPAARC are almost the same as RF. Still, after testing by setting each hyperparameter on each RF algorithm, SPAARC results is lower than RF. However, it produces pretty good accuracy. The training time is the longest. Random Forest and SPAARC algorithm methods are more superficial than MLP, so the training time is much faster than MLP. MLPs with more iteration settings will spend longer training or execution time, becoming the weakness of MLPs [26].

It can provide the best result because it is one approach of the ensemble method which combines several base models to produce one optimal predictive model. A large group of uncorrelated decision trees can produce more accurate and stable results than any individual decision tree. For example, in an ensemble method in a Random Forest, an increasing number of trees (J) can stabilize generalization error and converge surely to a limit [27]. Generalization error is related to measuring how accurately the algorithm can predict the outcome. Meanwhile, generalization error initially decreases in other ensemble methods as the number of trees (J) increases. When the number of trees (J) becomes too large, overfitting and generalization error increases.

In the RF model with specified hyperparameters and a high accuracy rate of 95.8%, the model also produces a good level of precision and recall. The precision (positive predictive rate) reached 97.1%, and the recall or sensitivity rate (true positive rate) reached 95.1%. Then the tested RF model can be used to create a predictive model to predict customer satisfaction with precision. It is also shown in Table 13.

Table 13. Confusion Matrix

	True (Satisfied)	True (Dissatisfied)	Precise (%)
Predicted (Satisfied)	61615	1850	97.100
Predicted (Dissatisfied)	3141	53005	94.400
Recall (%)	95.100	96.600	

CONCLUSION

Several machine learning algorithms were compared, Split Point and Attribute Reduced Classifier (SPAARC), Multilayer Perceptron (MLP), and Random Forest, to determine the classification model on passenger satisfaction. SPAARC was chosen in this study because the method was relatively new and had the advantage of short training time. All models produce an accuracy rate above 90% based on the results. However, the highest accuracy was obtained by the Random Forest method with the decision tree number hyperparameter setting 100 and max node size 100. The accuracy value generated in the random Forest model was 95.827%, and F-score was 0.958, and the training time was 84.53 seconds.

Random Forest can have the best performance because the ensemble method in Random Forest can stabilize generalization error and converge surely to a limit. This stabilized generalization error then results in better model accuracy. Further, this Random Forest modelling can be developed to identify features that make customers satisfied with the airlines and features that need improvement from dissatisfied customers.

REFERENCES

[1] M. Saut & V. Song, "Influences of airport service quality, satisfaction, and image on behavioral intention towards destination visit," *Urban, Planning and Transport Research*, vol. 10, no. 1, pp. 82-109, 2022, doi: 10.1080/21650020.2022.2054857

[2] S. S. Roy, D. Kaul, R. Roy, C. Barna, S. Mehta, and A. Misra, "Prediction of customer satisfaction using naive bayes, multiclass classifier, K-star and IBK," *Advances in Intelligent Systems and Computing*, vol. 634, pp. 153-161, 2018, doi: 10.1007/978-3-319-62524-9_12.

[3] S. Kumar and M. Zymbler, "A machine learning approach to analyze customer satisfaction from airline tweets," *Journal of Big Data*, vol. 6, no. 1, pp. 1–17, 2019, doi: 10.1186/s40537-019-0224-1.

[4] V. García, R. Florencia-Juárez, J. P. Sánchez-Solís, G. Rivera-Zarate, and R.

Contreras-Masse, "Predicting Airline Customer Satisfaction using k-nn Ensemble Regression Models," *Research in Computing Science*, vol. 148, no. 6, pp. 205–215, 2019, doi: 10.13053/rcs-148-6-15.

[5] K. Hulliyah, "Predicting Airline Passenger Satisfaction with Classification Algorithms," *IJIIS International Journal of Informatics and Information Systems*, vol. 4, no. 1, pp. 82–94, 2021, doi: 10.47738/ijiis.v4i1.80.

[6] D. Yates, M. Z. Islam, and J. Gao, "SPAARC: A fast decision tree algorithm," in *Communications in Computer and Information Science*, 996, Springer Singapore, 2019.

[7] S. Suthaharan, *A Cognitive Random Forest: An Intra- and Intercognitive Computing for Big Data Classification Under Cune Condition*, 1st Ed., vol. 35. Elsevier BV, 2016.

[8] S. Suthaharan, "Machine Learning Models and Algorithms for Big Data Classification: Thinking with Examples for Effective Learning," in *Integrated Series in Information System*, Springer, US, 2016, doi: 10.1007/978-1-4899-7641-3

[9] A. Fattah, M. M., P. S., and T. F., "A Decision Tree Classification Model for University Admission System," *International Journal of Advanced Computer Science and Applications*, vol. 3, no. 10, pp. 17–21, 2012, doi: 10.14569/ijacsa.2012.031003.

[10] D. Romahadi, F. Anggara, A. F. Sudarma, and H. Xiong,"The implementation of artificial neural networks in designing intelligent diagnosis systems for centrifugal machines using vibration signal," *SINERGI*, vol. 25, no. 1, pp. 87-100, 2021, doi: 10.22441/sinergi.2021.1.012

[11] W. Aribowo, "Elman-recurrent neural network for load shedding optimization," *SINERGI*, vol. 24, no. 1, pp. 29-36, 2020, doi: 10.22441/sinergi.2020.1.005

[12] A. Maceika, A. Bugajec, O. R. Šostak, and T. Vilutienė, "Decision Tree and AHP Methods Application for Projects Assessment: A Case Study," *Sustainability*, vol. 13, no. 10, ID: 5502; 2021, doi: 10.3390/su13105502

[13] Y. Y. Song and Y. Lu, "Decision tree methods: applications for classification and prediction," *Shanghai Archives of Psychiatry*, vol. 27, no. 2, pp. 130–135, 2015, doi: 10.11919/j.issn.1002-0829.215044.

[14] S. H. Jayady and H. Antong, "Theme Identification using Machine Learning Techniques," *Journal of Integrated and Advanced Engineering (JIAE)*, vol. 1, no. 2, pp. 123-134, 2021, doi: 10.51662/jiae.v1i2.24

[15] Y. Karaki and N. Ivanov, "Hyperparameters of Multilayer Perceptron with Normal Distributed Weights," *Pattern Recognition and Image Analysis*, vol. 30, no. 2, pp. 170–173, 2020, doi: 10.1134/S1054661820020054.

[16] G. I. Diaz, A. Fokoue-Nkoutche, G. Nannicini, and H. Samulowitz, "An effective algorithm for hyperparameter optimization of neural networks," *IBM Journal of Research and Development*, vol. 61, no. 4, pp. 1–20, 2017, doi: 10.1147/JRD.2017.2709578.

[17] U. Cavusoglu, "A new hybrid approach for intrusion detection using machine learning methods," *Applied Intelligence*, vol. 49, no. 5, 2019, doi: 10.1007/s10489-018-01408-x

[18] J. L. Speiser, M. E. Miller, J. Tooze, and E. Ip, "A comparison of random forest variable selection methods for classification prediction modeling," *Expert Systems with Applications*, vol. 134, pp. 93–101, 2019, doi: 10.1016/j.eswa.2019.05.028.

[19] P. Probst, M. N. Wright, and A. L. Boulesteix, "Hyperparameters and tuning strategies for random forest," *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 9, no. 3, pp. 1–15, 2019, doi: 10.1002/widm.1301.

[20] M. J. Hartmann and G. Carleo, "Neural-Network Approach to Dissipative Quantum Many-Body Dynamics," *Physical Review Letters*, vol. 122, no. 25, p. 250502, 2019, doi: 10.1103/PhysRevLett.122.250502.

[21] M. K. Tiwari, R. C. Deo, and J. F. Adamowski, "Short-term flood forecasting using artificial neural networks, extreme learning machines, and M5 model tree," *Advances in Streamflow Forecasting*, pp. 263-279, 2021, doi: 10.1016/B978-0-12-820673-7.00012-3

[22] B. Zhu and Y. Liu, "General Approximate Cross Validation for Model Selection: Supervised, Semi-supervised and Pairwise Learning," *Proceedings of the 29th ACM International Conference on Multimedia*, October 2021, pp. 5281–5289, doi: 10.1145/3474085.3475649

[23] E. P. Cynthia and E. Ismanto, "Metode Decision Tree Algoritma C.45 Dalam Mengklasifikasi Data Penjualan Bisnis Gerai Makanan Cepat Saji," *JURASIK (Jurnal Riset Sistem Informasi dan Teknik Informatika)*, vol. 3, no. July, p. 1, 2018, doi: 10.30645/jurasik.v3i0.60.

[24] W. Baswardono, D. Kurniadi, A. Mulyani,

and D. M. Arifin, "Comparative analysis of decision tree algorithms: Random forest and C4.5 for airlines customer satisfaction classification," *Journal of Physics: Conference Series - IOPscience*, vol. 1402, no. 6, 2019, doi: 10.1088/1742-6596/1402/6/066055

[25] Y. M.Ganushchak, P. J. C.Barenburg, J. G. Maessen, and P. S. Nia, " ROC-Tree Algorithm for Stratification of Binary Classifier Sets with Varied Discrimination Threshold," *Advances in Bioengineering & Biomedical Science Research*, vol. 5, no. 3, pp. 113- 126, 2022.

[26] I. T. A. Nur, N. Y. Setiawan, and F. A. Bachtiar, "Perbandingan Performa Metode Klasifikasi SVM, Neural Network, dan Naive Bayes untuk Mendeteksi Kualitas Pengajuan Kredit di Koperasi Simpan Pinjam," *Jurnal Teknologi Informasi dan Ilmu Komputer (JTIIK)*, vol. 6, no. 4, p. 444, 2019, doi: 10.25126/jtiik. 2019641352.

[27] A. Antoniadis, S. Lambert-Lacroix, and J. Poggi, "Random forests for global sensitivity analysis: A selective review," *Reliability Engineering & System Safety*, vol. 206, ID: 107312, 2021, doi: 10.1016/j.ress. 2020.107312