2023 AIAA Aviation and Aeronautics Forum and Exposition
(AIAA AVIATION Forum), 12-16 June, San Diego, USA. Paper AIAA 2023-4326
DOI:10.2514/6.2023-4326

A Hybrid Ensemble Machine Learning Approach For Arrival Flight Delay Classification Prediction Using Voting Aggregation Technique

Desmond B. Bisandu[1*], Irene Moulitsas[2]

Cranfield University, Bedford, MK43 0AL, United Kingdom

I. Nomenclature

$\mu(i)$	=mean
c_j	=samples class label
$d(i)$	=discriminant power
d_i	=discriminant power
F	=features vector
$f_{i,j}$	=index of ith learning sample in the jth feature
F_{re}	=feature group at the rear
F_{top}	=features group at the top
M	=dataset matrix definition
m	=number of dataset sample
n	=number of dataset features
n_F	=number of the sample feature
$p(i)$	=discriminant class
S	=features significant
s	=standard error
X_i	=vector collection of features
α *and* β	=parameter of the filter feature significant
$\sigma(i)$	=standard deviation

II. Introduction

Many factors, such as safety, security, air carrier, maintenance, National Aviation System (NAS), weather and airport scheduling, affect flight plans in the civil aviation transportation processes [1-4]. Frequently, scheduled flights cannot arrive on time, affecting subsequent flights. Flight delay occurs conventionally if the flight is 15 minutes later than the scheduled departure or arrival time. Air traffic demand-capacity imbalances in arrival/departure flights resulting from continuous air traffic growth with limited airport expansion capabilities. For example, in 2017, there was 4.3% on European flights relative to 2016 [3]. It corresponds to an average of 1191 flights per day. Also, 20.4% of flights in 2017 experienced arrival delays of 15 minutes and above [6-8]. It is important to predict flight delays to improve airport efficiency and operation coordination. These delays have caused more damage to the aviation industry and passengers with almost an exponential increment, leading to poor passenger satisfaction and a loss of at least $20.5 million to flight delay-related issues in the United States in 2018 [5,9,10].

[1]*Ph.D. Candidate, Centre for Computational Engineering Sciences; desmond.bisandu@cranfield.ac.uk. AIAA Student Member.
[2]Senior Lecturer, Centre for Computational Engineering Sciences; i.moultsas@cranfield.ac.uk.
[1*,2]Machine Learning and Data Analytics Laboratory, Digital Aviation Research and Technology Centre (DARTeC).

With the rapid increase in the amount of data from the air transportation industry due to the constant development in the sector, processing the huge amount of data is becoming tedious and almost impractical with only traditional processing methods such as balancing and shuffling. Flight delay

prediction based on machine learning has achieved good performance in the past. Some methods such as KNN, decision tree and random forest [3], Support vector classifier, and other methods have good accuracy on large flight on-time datasets but are slow in computation compared with deep learning methods. Machine learning superior performance in prediction compared with other traditional statistical methods has been massively recorded in artificial intelligence and transportation-related research [11,12,13,14,15]. Support Vector Machine (SVM) is among the most popular machine learning techniques for classification problems because of high prediction results, especially with high dimensional and imbalanced datasets [13,16,17]. However, due to the high dimension of the feature, good pre-processing is required to perform flight delay predictions. For instance, one-hot encoding processing will be needed for flight landings and take-off features, leading to sparse data. Encoding aircraft tail coding, airlines etc., may also be required. This encoding process removed much information that may affect the model's performance output.

In recent years, machine learning and deep learning have become the most popular and acceptable methods to predict flight delays [8,18]. Many researchers have proposed techniques for flight delay predictions using multiple perspectives and uncertainty factors affecting the flight's operation. For instance, using a cause-and-effect test, Wen-BoDu [18] establishes a delay cause-and-effect network (DCN) by analysing the factors that affect flight delays. Alice Sternberg [19] studied hidden flight delays using association rules. Also, Bin Yu [16] used a deep belief network to analyse and predict internal flight delay patterns. Poornima [17] applied a non-parametric reinforcement learning method (RL) to study the actual time of aircraft and taxi time deviation. Rodriguez-San et al. [18] Cheng-Lung Wu et al. [19] use Bayesian network and reliability models to predict flight delays. Roberto Henriques [20] uses a multilayer perceptron to predict flight delay. Sun Choi et al. [21] built a model for flight delay prediction based on Ada-boost with sample imbalance flight data set. The result of the forecast was relatively good.

Ensemble learning methods, such as the Boosting model, have improved the accuracy of weak classification algorithms. A Series of prediction functions are constructed to reduce the loss continuously through a series of iterations by the model. Cat-boost is an open-source algorithm for gradient enhancement developed by Dorogush [22]. The users are processing classification characteristics from a large dataset, which can solve a ranking problem, classification and regression, especially in the imbalance dataset prediction, which has superior performance than the XG-boost and Light-GBM as shown in previous studies [22,23]. Cat-boost applies the symmetric tree method in processing categorical features, which has an advantage over XG-boost in improving the model's performance and robustness with a high accuracy rate.

Our contribution is summarised as follows: Firstly, we proposed a hybrid ensemble learning model based on hard voting techniques for predicting arrival flight delay with a feature sub-sampling technique. Secondly, our proposed methodology generalisation relies on a relative performance comparison between the strategic schedule of flight delay prediction on different well-known ensemble machine learners. Thirdly, we provide experimental evidence of our proposed method results compared with Boosting and stacking approaches. We clearly show the improvement in our proposed method.

The rest of the paper's organisation is as follows. Section II presents the introduction. Section III contains an overview of some most important related works. The materials and methods are presented in Section IV. Section V presents the results and discussion. Finally, Section VI presents the conclusion.

III. Related Works

Any disruption in the air transportation system processes, aside from the Traffic Flow Management (TFMs), including but not limited to a mechanical problem, severe weather conditions, an outage at the origin/destination airports, or staff shortage, may result in a flight delay. Air transportation delay occurrences may be at different flight phases: en-route, arrival and departure and can affect passenger schedules, crew and flight (e.g., missing a connecting flight). Also, delay propagation can affect arrival and departure flights [23,24]. Therefore, the ability to control the factors and predict flight delays is an essential objective for airlines.

Flight delay analysis and prediction have been addressed extensively in the literature, as seen in [25,26,27]. A data-driven model for flight delay distribution estimation was proposed by [25]. Flight statistics were determined for ten major US airports through a historical dataset. Departure and arrival delays have been modelled based on these statistics as normal and Poisson processes, respectively [28]. The historical data from Beijing Capital International Airport was used to estimate the probability density function of departure and arrival delays of flights with an optimal generalised extreme value model [27]. The authors in [29] proposed a statistical model that estimates departure flight delay distributions and seasonal patterns using the Denver International Airport. Random residuals' seasonal and daily propagation patterns were considered. In [30], a flight delay detection pattern based on a frequency

analysis at Orlando International Airport was proposed to analyse airport network propagation delays. The authors used a Bayesian network for the network queue in [31].

Authors in [32,33] proposed machine learning approaches for flight delay classification with a prediction horizon of several days before the flight execution occurs. In [32], the authors achieved an accuracy rate of 0.268 with five days of weather forecast available before the flight's execution day. A random forest classifier with exclusively weather-related feature training was employed. Authors in [33] used unfavourable weather conditions to exclusively propose a model for flight delay classification. A balanced dataset was used with a random sampling algorithm to decrease delayed samples based on the under-sampling algorithm. The features considered were the origin/destination airport, weather conditions, and scheduled departure/arrival time. An accuracy rate of 0.86 is achieved with a recall of 0.87 on a threshold of 60 minutes (i.e., any flight with 60 minutes or more time relative to arrival time is considered delayed) by the random forest method. A Receiver Operating Characteristics (ROC) of XGBoost achieved was 0.53 for prediction in 20 Asian airports for a low-cost airline. From the literature results, ensemble methods can generate robust results for classifying datasets with noise having an improved prediction than a single classifier [34,35,36].

Additionally, there are many advantages of using ensemble learners; multi-classifier systems have proven to be the potential for developing robust learning methods with noisy data in both class labels, missing values and features. Classifier results can have increased variance due to noisy training data. However, the variance can be reduced by learning the combined decision of a committee of hypotheses. In particular, bagging has shown outstandingly high noise levels as a variance-reducing method.

Despite all the efforts to find universal solutions to the problem of flight delay prediction through the various contributions from academia and the industries, It has been reiterated that there is a need to develop further hybrid models that can effectively and efficiently apply a repeatable and systematic approach to identify and predict flight delays from different viewpoints. Identifying and measuring the factors that significantly affect flight delays is important to the aviation community and cannot be overemphasised. Because it will help controllers, traffic managers, and airlines proactively adjust and minimise operational impacts. Also, identifying each factor's robust occurrence probability in the delay scenarios will guide airlines and other stakeholders in the planning process and disruption recovery. Thus, we proposed a hybrid ensemble machine learning method based on the voting aggregation technique for arrival flight delays for non-weather impacted flight delays.

IV. Materials and Methods

In this research, we proposed an ensemble members construction approach by learning from the sub-set samples of features. The proposed method is shown in Figure 1 and can be summarised in the following stages:

Stage 1: Feature sub-sampling: The sample features will produce accurate and diverse classifiers. A hybrid sub-sampling method is applied for the feature sub-sampling.

Stage 2*:* Candidate classifiers generation: Machine learning methods are applied to generate a pool of candidate classifiers. The input for each classifier is a subset of the features generated from stage 1. Support Vector Classifier (SVC) is chosen as the machine learning algorithm for this study's base classifier. SVC application is due to its ability to deal with high dimensional training data features.

Stage 3*:* Ensemble Committee Construction: Two sub-steps are involved. The first is the base classifier behaviour characterisation generated. The other is fusing the voting aggregation of the classifiers that construct a robust committee for classification. The high diversity of the classifiers means classifiers disagree with each other but have good classification accuracy as committee members. The majority voting mechanism is then applied to make the final decision for the classification ensemble.

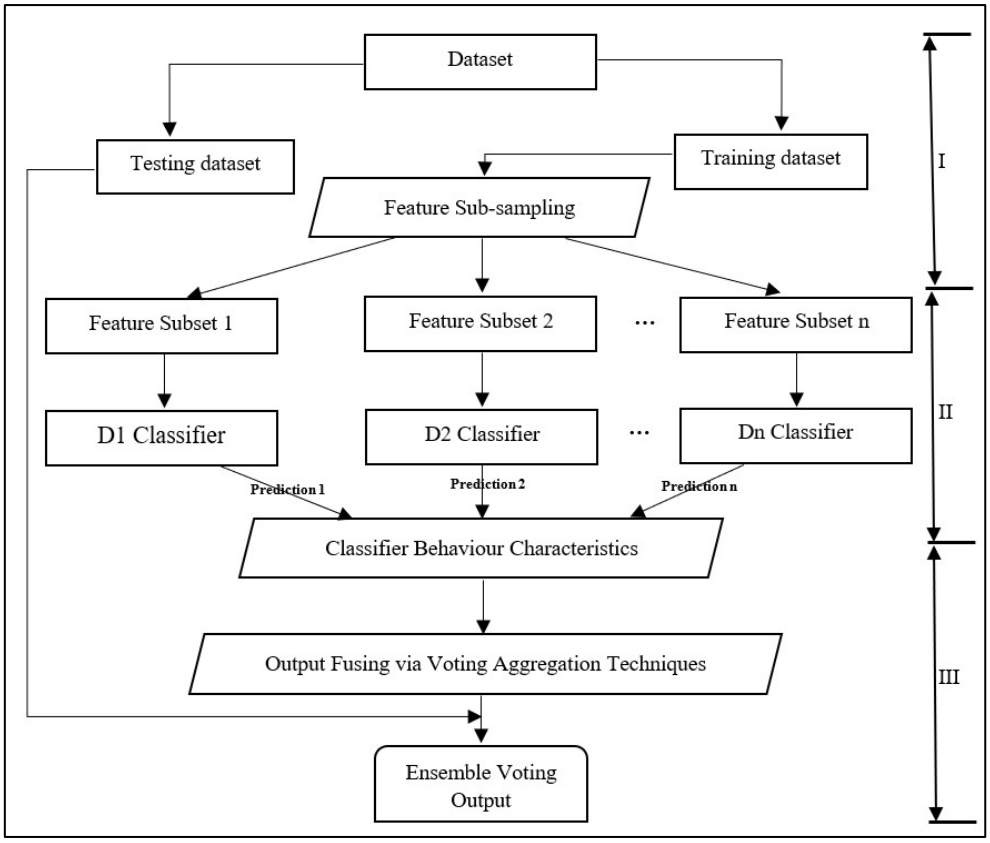

Figure 1: The proposed ensemble method

A. Feature Characterisation

We utilised a hybrid feature selection algorithm to select features to help produce accurate and diverse classifiers. Our approach uses a random selection of features to construct the base classifiers from the original dataset. The randomise selection method makes the ensemble classification model less sensitive to feature selection.

Assuming a dataset for flight on time is represented by a matrix M is $m \times n$ and M = $[f_{ij}]_{m\times n}$ where m is the number of learning samples, and n is the number of features. An instance (Matrix M row) is an expression of a vector $X_i = (f_{i1}, f_{i2}, \dots, f_{in})$, where f_{ij} represents the index of ith learning sample in the jth feature. Every sample is labelled with a class $c_j = \{-1, +1\}$. Assuming we have learning instances from $j = 1$ to k are for class +1 and learning sample from $j = k + 1$ to n are for class -1, the ith feature vectors expression for class +1 and -1 are expressed as $X(i,+) = \{f_{i1}, f_{i2}, \dots, f_{in}\}$ and $X(i,-) = \{f_{i(k+1)}, f_{i(k+2)}, \dots, f_{in}\}$, respectively.

The feature's significance is important because it can affect the performance of the individual base classifiers and is measured by the features' discrimination power. Characterisation of the feature's discrimination power measures the difference between the different groups' expression levels of learning samples [37]. Equation (1) shows the discriminant power.

$$d(i) = \frac{diff\big(X(i,+),X(i,-1)\big)}{l_0+l}, \quad (1)$$

Where $diff\big(X(i,+),X(i,-1)\big)$ is the group difference between $X(i,+)$ and $X(i,-)$, $l_0$is a regularisation constant, and s is the standard error.

Measuring features' discrimination power is presented in [38] to select desired features for classification. For a feature in the ith, let $[\mu_+(i), \sigma_+(i)]$ and $[\mu_-(i), \sigma_-(i)]$ represents the means and standard deviation of the learning sample expression in class +1 and -1, respectively. As shown in Equation (2) and Equation (3), and Equation (4) and Equation (5), respectively.

$$\mu_+(i) = \frac{1}{k}\sum_{j=1}^{k} f_{ij}, \quad (2)$$

$$\sigma_+(i) = \sqrt{\frac{\sum_{j=1}^{k}(f_{ij} - \mu_+(i))^2}{k}} \quad (3)$$

and

$$\mu_-(i) = \frac{1}{n-k}\sum_{j=k+1}^{n} f_{ij}, \quad (4)$$

$$\sigma_-(i) = \sqrt{\frac{\sum_{j=k+1}^{n}(f_{ij} - \mu_-(i))^2}{n-k}} \quad (5)$$

The measure of the ability of a feature $X(i)$ discriminate class $+1$ from -1 is expressed as in Equation (6).

$$p(i) = \left|\frac{\mu_+(i) - \mu_-(i)}{\sigma_+(i) - \sigma_-(i)}\right| \quad (6)$$

where the capability of the associated feature for distinguishing the classes of the learning sample is measured by $d(i)$ or $p(i)$. Large values of $d(i)$ or $p(i)$ It means the associated feature's expression level is highly differential between the two classes $+1$ from -1.

B. Sub-sampling Method

Assume we have a set of features $F = \{f_1, \ldots, f_n\}$ and the feature is associated with an assigned significance value, i.e. $S = \{s_1, \ldots, s_n\}$ where Equation (1) or Equation (6) is used to calculate the $s_i = d(f_i)$ or $p(f_i)$, the hybrid sub-sampling algorithm is in four stages to generate a subset of the feature ($\hat{F}$):

1) Parameter calculation and feature filtering.
2) Feature ranking.
3) Feature partition.
4) Feature sub-sampling and recombination.

In the first stage, the features are filtered with their levels of significance ($d(f_i)$ or $p(f_i)$) are calculates the parameters according to the number of sample features (n_F), the two parameters (α and β) and less than a given threshold (γ):

1) The top group feature size (Denoted as F_{top}):
   $$n_{top} = \alpha \times n_F \quad (7)$$
2) Number of features sampled from F_{top}
   $$n_1 = \frac{n_{top}}{\beta} \quad (8)$$
3) Number of features selected from the group of the remaining features.
   $$n_2 = n_F - n_1 \quad (9)$$
   For instance, if we have $n_F = 25$, and the parameters $\beta = 4$ and $\alpha = 0.4$ then the size of F_{top} is $n_{top} = 10, n_1 = 2.5$ and $n_2 = 22.5$.

The features are ranked according to their significance and partitioned into two sets in the second stage. The top n_{top} features are contained in one set (Top significance features denoted as F_{top}) and the remaining features contained in the other set (denoted as F_{re}). The feature sub-sampling is performed using two different approaches from the two feature sets. The top feature F_{top}, there is an equal probability of sampling because all the members have good discrimination power, while for the features in F_{re} , there are different probabilities among each of the samples. The sampling is performed using random sampling. For the F_{re} In sampling, the significance of the features determines the probabilities of the sampling, i.e., a higher significance (denoted by $d(f_i)$ or $p(f_i)$) the feature has a higher sampling probability. The features are sampled in the end, respectively, from F_{top} and F_{re} are combined to generate the subset of one sample. Thus, sub-sample are run k-times, and then a sub-set of features produced denoted by $\widehat{F_k}$.

$$\widehat{F_k} = F'_{top} - F'_{re} \quad (10)$$

The significance of the feature can be normalised with Equation (11).

$$\widehat{s_i} = \frac{s_i}{\sum_{Fre}^{n} s_i} \quad (11)$$

The cumulative significance of the feature distribution function can be calculated using Equation (12).

$$cd_i = \sum_{j=1}^{i} \widehat{s_j} \quad (12)$$

C. Support Vector Classifiers

Many researchers have applied support vector classifiers (SVC) for different classifications and predictions with promising results. SVC minimises the objective function (the mean absolute or mean square error) by finding the hyperplanes producing the largest decision function value separation between the instances at the borderline of the two classes. However, the other modelling methods minimise the objective function of whole training instances [39].

Given data with label $D = \left\{\left(x_j, y_j\right)_k\right\}$ where, j = 1, 2, …, n; k = 1, 2, …, m, the construction of a decision function $f(x)$: $R^n \longrightarrow R$ is the SVM-based flight delay classification target, such that for every x_j, yields $f\left(x_j\right) > 0$ for y_j = +1, and $f\left(x_j\right) < 0$ for $y_j = -1$. A linear decision function $f(x) = W^T + b$ or a non-linear decision function $f(x) = W^T \text{ø}\,(x) + b$, where the non-linear transform function is $\text{ø}\,(x)$ is employed by the SVC-based classifier. The $f(x)$ is determined by minimising $J\,(w, \varepsilon) = \frac{1}{2}\,\|W\|^2 + C\sum_{j=1}^{l} \varepsilon_j$ subjected to $y_j(W^T x_j + b) \geq 1\text{-}\,\varepsilon_j$ (linear) or $y_j(W^T \text{ø}(x_j) + b) \geq 1\text{-}\,\varepsilon_j$ (non-linear), where the regularisation parameter is represented by C >0 and the slack parameters are $\varepsilon_j \geq 0$ (j = 1, 2, 3, …, l).

For minimising the $J\,(w, \varepsilon)$, the linear combination of $\text{ø}(x_j)$ can be the expression of W such that $w = \sum_{j=1}^{l} a_j y_j\, \text{ø}(x_j)$. When we substitute w into $f(x) = W^T \text{ø}\,(x) + b$ will result into $f(x) = \sum_{j=1}^{l} a_j y_j\, K(x_j, x) + b$, where the kernel function is represented by $K(u, v) = \text{ø}^T(u)\text{ø}(v)$. Then $a_j \geq 0$ (j = 1, 2, …, l) can be obtained from minimising $W\left(a_j\right) = \sum_{j=1}^{l} a_j - \frac{1}{2}\sum_{j=1}^{i}\sum_{k=1}^{l} a_j a_k y_j y_k K(x_j x_k)$ with the condition $0 \leq a_j \leq C$, and $\sum_{j=1}^{l} a_j y_j = 0$. SVC utilises three major kernel functions for classification are Linear $K(u, v) = uv$, polynomial $K(u, v) = (uv + 1)^d$ and the Gaussian function $K(u, v) = \exp\left(-\frac{\|u-v\|^2}{2\rho^2}\right)$. Where u, v are the features and their respective labels for the function K; d is the set of polynomial degrees, while ρ is the length scale.

D. Boosting and Bagging Ensemble Machine Learners

These ensemble learning methods used the instance re-sampling scenario. The boosting algorithm applied in this research is AdaBoost, designed by the authors in [39]. A minimised weighted error is invoked on each iteration to AdaBoost by the learning algorithm on the training set and generates the base classifier accordingly. The bagging algorithm on each learning algorithm with a training set consisting of m training examples drawn from the random selection of the original dataset with replacement [4,32,40,41]. The training examples' weights are adjusted according to the classification weighted error of the developed classifier model. The misclassified training examples are assigned more weights, while the training examples that were correctly classified are assigned fewer weights. A classification margin is induced in bagging by the random sampling procedure.

Additionally, the diversity in the resulting classification margin of bagging is due to the difference in feature subspace performance, and an ensemble learner needs to be accurate. It studies the average bagging error and then converts it to an optimisation problem for feature weight determination. The weights are assigned to the subspaces using a randomised technique in classification construction. The final classifier is constructed by individual classifiers voting for Boosting and Bagging [9,42,43].

E. Prediction Based on Proposed Voting Ensemble Method

We employed the majority voting method for making the final decision based on base classifiers' outputs. Given c_c classifiers $C_i\,(x) : R^n \rightarrow \{-1, +1\}$, a weight $w_{i\in}[0, 1]$ is assigned to each classifier to show its significance. Assuming a new instance x, for each of the class predicted $C_i\,(x) \in \{-1, +1\}$ for i = 1 - c_c, the majority vote method generates the final classification. Equation 13 represents the ensemble method of majority voting.

$$C_{ens} = \text{sign}\left(\sum_{i=1}^{c_c} w_i c_i\ (x)\right) \quad (13)$$

In ensemble methods, the diversity of the ensemble is crucial for the success of the ensemble classifiers, which is the disagreement of the base classifiers [37]. A set of base classifiers is ideally selected to maximise accuracy and diversity to help seek a robust ensemble classifier. The effectiveness of compact ensemble committee selection by feature selection method has been proven by much research [36,37,44]. The proposed method in this paper characterises the behaviour of the candidate base classifier before the appropriate base classifiers are selected to construct a high-performance classification committee. Behaviours candidate base classifiers are characterised.

(a) Base classifiers are grouped in terms of their classification behaviour.
(b) An appropriate subset of the base classifiers is selected to construct the classification committee.

The behaviour of the base classifiers associate feature subset). Selecting the best classifiers from different committees is based on the base classifier with the best accuracy from each committee to ensure individual committee members' accuracy. The performance is similar in selecting the diverse classifiers from the candidates.

F. Dataset Description and Pre-processing

A flight that arrives or departs 15 minutes or more is considered delayed. In comparison, punctual or on-time flights are the ones that arrive or depart within the scheduled time, according to IATA [45,46,47,48]. In this study, we employed data from the US commercial flight operation downloaded from the United States Department of transportation for the month of January to March 2013, with 1,048,576 instances, including all the missing variables, NA and inconsistencies record [49]. The provided features include airport information, plans; landing city; aircraft information; schedule and actual landing information summing up to 21 features from the raw dataset after eliminating all records with too much noise and at least four missing values. Some categorical variables, such as flight date, carrier information and aircraft tail number, cannot be processed directly in the flight delay dataset. Variables such as integer and floating-point data can be calculated directly. We then divided the dataset into training and testing sets with 70% and 30% ratios to train the model and predict the flight delay using the proposed algorithms.

G. Feature Selection

The role of feature choice cannot be overemphasised because it plays a key role in the model's final results. Different features are selected for suitable models according to their characteristics to achieve better results. The embedded feature selection method uses the machine learning model for feature selection. The feature selection process integrates the learner training process and uses the feature score obtained during the training process to select features automatically.

Because ensemble learners such as random forest can score features, it makes it possible to select parameters of more importance to the model based on the ranking generated from the model's training, even with an unknown threshold. Therefore, we used the random forest to evaluate and select each feature, and we finally obtained the following 15 features for training the model. Some of the features have a strong correlation with each other. They need to be removed before the evaluation, for example, the actual elapsed time of an aircraft and other timing features that indicate a delay of an airline. Because flight delay is subject to actual departure or arrival delay, the categorical variables 0 and 1 are used as objects of prediction, where 0 means on-time flight performance and 1 means delayed flight. Table 1 contains the list and description of this specific type of employed features in this research.

Table 1: Dataset features and description

Features	Data Type	Description
Arr_Delay	int64	Difference between scheduled and actual arrival time
Dep_Delay	int64	Difference between scheduled and actual departure time
Arr_Time	float64	Arrival time
Dep_Time	float64	Departure time
Taxi_In	int64	The number for an aircraft taxi coming in
Taxi_Out	int64	The number for an aircraft taxi going out
Actual_Elapsed_Time	float64	Elapse time
CRS_Elapsed_Time	float64	Schedule elapsed time
CRS_Arr_Time	float64	Scheduled arrival time
CRS_Dep_Time	float64	Schedule departure time
Day_of_Month	int64	Day of the particular month
Flight_Num	object	A unique number of the flight
Air_Time	float64	Time of flight in minutes
Dep_Del_Label	int64	Category flight delay and on time
Distance	int64	Different airport distances (miles)

H. Model Performance Evaluation and Validation

The criteria for evaluating and validating every machine learning model are important for the final result measurement. There are different focuses for different scenarios. Generally, all classifier's pros and cons can be evaluated using precision, recall, F1-score and accuracy based on some objective functions or loss and cost

function for convergence. The proportion of all correctly classified or judged positive samples is called accuracy. The proportion of the positive samples is known as the recall ratio because the flight time probability of arrival flight is far more than the probability of the flight delay. In this research, we pay more attention to on-time accuracy, precision and accuracy. At the same time, model accuracy must be higher than the accuracy of the sample containing the majority of samples so that the model's efficiency is better than the result. A classification reference that ensures prediction accuracy is higher than classified with the largest proportion of all the variables to be explained into a category. Table 2 shows the confusion matrix for a given prediction problem.

Table 2: The Confusion Matrix

		Predicted Class	
		Positive	Negative
Actual Class	Positive	True Positive (TP)	False Negative (FN)
	Negative	False Positive (FP)	True Negative (TN)

The selection of a decision threshold in a binary classification problem influences the classifier's quality. It is common to determine the decision threshold through the ROC-AUC and FPR-FNR plots, respectively. The receiver operating characteristic (ROC) curve depicts the same signal stimulus susceptibility on each point. The negative-positive class rate is the horizontal axis representing the true negative sample division into positive samples. In contrast, the vertical axis is the accuracy rate, and the area formed by the ROC is the AUC and the x-axis. The probability of the incorrect classification of the positive or negative samples when taking different thresholds is given by the FPR-FNR.

According to the airline's on-time statistics data and unbalanced data characteristics of the samples for binary forecast requirements, AUC-ROC and F1-score were selected as the pros and cons of the model. The evaluation criteria for the model are explained as follows in Equations 14 to Equation 19:

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN} \quad (14)$$

$$\text{Precision} = \frac{TP}{TP+FP} \quad (15)$$

$$\text{Recall} = \frac{TP}{TP+FN} \quad (16)$$

$$\text{FPR} = \frac{FP}{FP+TN} \quad (17)$$

$$\text{FNR} = \frac{FN}{TP+FN} \quad (18)$$

$$\text{F1} = \frac{2*Precision*Recall}{Precision+Recall} \quad (19)$$

The harmonic average of the precision and recall are the F1-score having a value ranging as F1∈ [0,1], and the optimal is when the model is 1. For the k-Folds cross-validation, we employed stratified cross-validation (SCV) [50].

V. Results and Discussion

This section discusses the results and findings from the experiments conducted in this study. We conducted our experiment on a Personal Computer (PC) with an Intel (R) Core(TM) i7-9700 CPU with a processor speed of 3.00GHz and 32GHz RAM. We used libraries such as TensorFlow Core-2.4.1, TensorFlow GPU-2.4.1, NumPy-1.19.1, pandas-0.25.3, sci-kit learn-0.23.2, Scipy-1.5.2, PySimpleGUI-4.29.0, seaborn 0.11.2 and Matplolib-3.3.1.

Our experimental study has a principal objective of verifying the effectiveness of applying ensemble machine learning on flight datasets by comparing the performance of the proposed ensemble method and other widely used ensemble learners (Bagging, Boosting, Simple Averaging e.t.c.). The performance comparison has been performed based on common conditions of the experiment, with additional data pre-processing performed before constructing the base classifier committee. The results from the experiment have shown that in conventional machine learning techniques (boosting, bagging and single classifier), there is a significant and consistent improvement in accuracy, precision and recall produced by our proposed method under different evaluation benchmarks.

Furthermore, there are competitive chances between the obtained results from this study and the methods existing in the literature. In [42] and [12], a boosting ensemble machine learning method was proposed, and the

best accuracy of 80.44% and 87.72% were obtained for the arrival flight on-time dataset, respectively. While in [3], a stacking ensemble machine learning for departure and arrival flight delay prediction was compared, and the best accuracy of 82.22% was obtained for departure and arrival flight delay predictions.

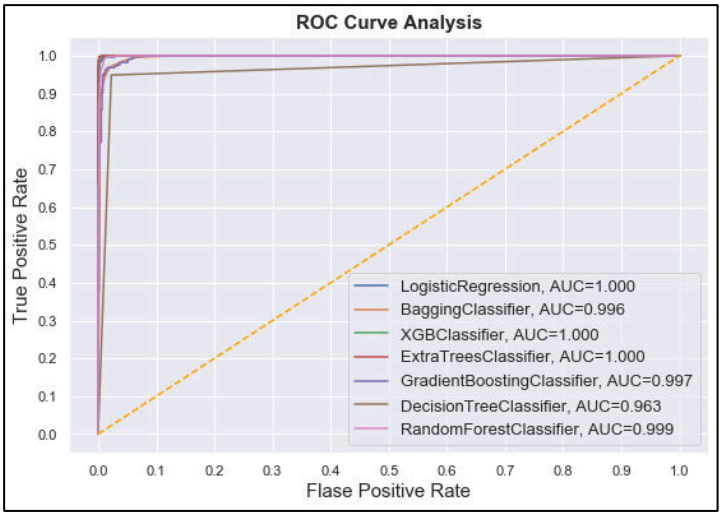

Figure 2: ROC Curve of models

Figure 2 shows the ROC (receiver operating characteristics) curve that measures the generalisation of the algorithm's ability. The AUC (under the curve) is the area under the ROC [3,51]. An algorithm is considered better if it is closer to 1. We calculated and plotted the ROC of each base committee algorithm and their respective AUC scores, as shown in Figure 5. The Logistic Regression, Extra Gradient Boosting and Extra Trees reach 1.000, followed by the Random Forest with 0.999. The result of Bagging and Gradient Boosting was 0.997 and 0.996, respectively. The Decision Tree has 0.996, and it is the smallest compared to the base classifiers; this means that all the algorithms with lower values contribute less concerning the overall performance of the ensembles on the prediction of the arrival delay. Something worth noting is if there is an impact on the ensemble accuracy based on the committee algorithms' performance. As seen from the classifiers, the AUC are above 80%. It indicates the good overall sensitivity of the classifiers to predict flight delays. We assigned special names to all the ensembles, as shown in Table 3.

Table 3: Special names for ensembles

S/No	Ensemble Architecture	Special Name
1	Simple Averaging Ensemble Model	SAEM
2	Boosting Ensemble Model	BOEM
3	Stacking Ensemble Model	SEM
4	Hybrid Voting Ensemble Model	HVEM

The names proposed in Table 2 are not formal but a conventional approach for our convenience. It is to help us minimise long titles in our plots.

Table 4: Prediction results of different methods

S/No	Metrics	Methods			
		SAEM	BOEM	SEM	HVEM
1	Precision	0.78	0.81	0.93	0.97
2	Recall	0.67	0.80	0.94	0.98
3	Accuracy	0.67	0.80	0.95	0.99
4	F1-Score	0.58	0.78	0.90	0.97

Table 4 shows the ensemble algorithms comparison. We find that the HVEM model has better performance with an accuracy of 0.99, precision of 0.97, recall of 0.98 and F1-score of 0.97, respectively, followed by SEM, having an accuracy of 0.95, recall of 0.94, and precision of 0.93 F1-score of 0.90, respectively. The BOEM is the next ensemble with a performance of 0.81 precision, 0.80 on recall and accuracy while having a 0.78 F1 -score. The SAEM performed poorly, with accuracy and recall of 0.67, a precision of 0.78 and F1-score of 0.58. We run the

experiment on all the models with the same number of base algorithms to help us understand the impact of the ensemble classifiers on the results of the prediction. The high performance observed on the HVEM could be due to the ability of the committee base algorithms to effectively manage the trade between the stronger and the weaker learners in deciding the best results among the individual committee. The overall accuracy of the ensemble indicates great stability of the HVEM, as seen in Figure 3 with all other metrics plots. As we have mentioned, there is no "greatest algorithm" or a "multipurpose algorithm" in the machine learning field. Therefore, the HVEM algorithm could be a good solution for selecting algorithms, especially complex and enormous datasets like flight datasets.

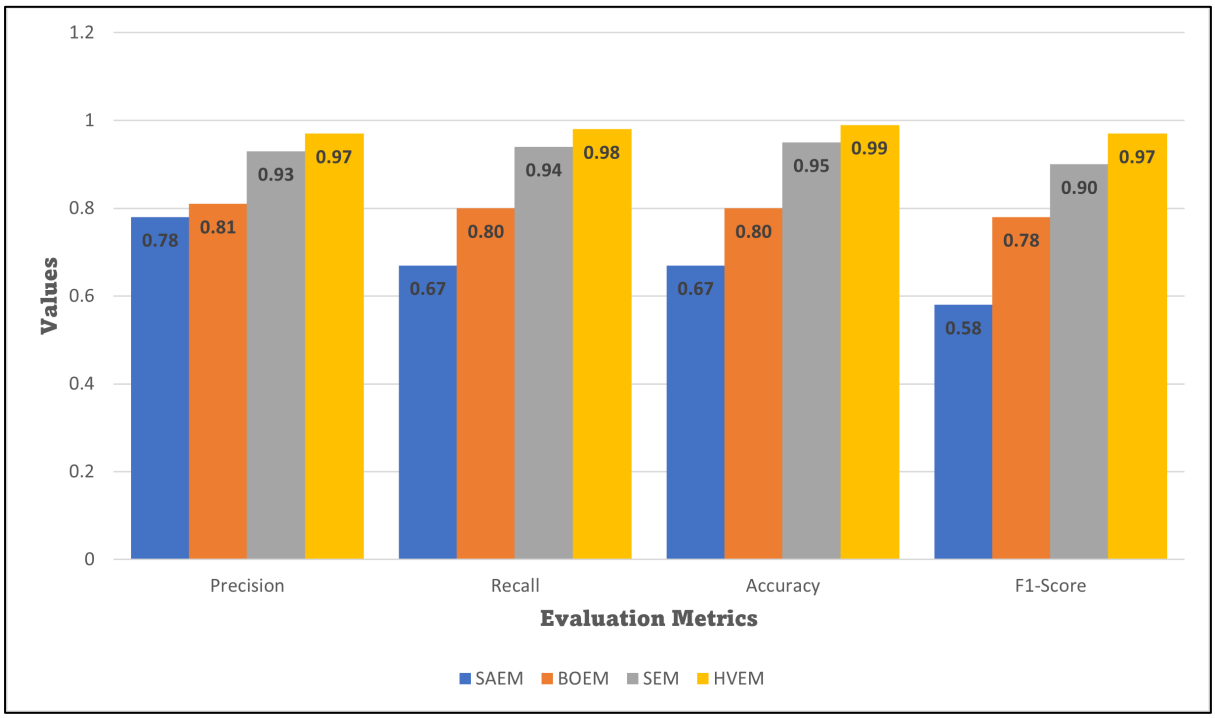

Figure 3: Prediction results plot of different methods

Table 5 shows the results of the cross-validation of the bagging ensembles. We performed 50-Folds cross-validation across the classifiers committee to understand how well the classifiers understand new instances of the flight delay task. After 50-fold validation, the Support Vector classifier and Extra Tree outperformed other models with an accuracy value of 0.99. At the same time, Random Forest and Decision Tree had a value of 0.98, and Logistic Regression had 0.62. This variation in the results is due to how each sub-sample is assigned to the fold.

Table 5: Results of Bagging Ensemble Models with 50-Folds Cross-validation

Bagging Methods	Special Name	Mean Cross-validation results (K-Folds = 50)
Logistic regression model	LogReg	0.62
Decision tree model	DecTree	0.98
Random forest model	RanFor	0.98
Extra trees model	ExtTree	0.99
Support Vector Classifier model	SupVecClas	0.99

VI. Conclusion

The major concern for the airports, airlines, passengers and other stakeholders is reducing or eliminating flight delays. However, it is not an easy task to minimise delay time. Therefore, arrival delay prediction turns out useful. Several researchers have tried developing flight delay prediction models with good precision and accuracy. Our study proposed a hybrid ensemble learning approach using voting aggregation based on supervised learning. After performing the data stage pre-processing, the base classifiers were used to train the model. The experiment showed the highest accuracy score of 0.99 for the proposed hybrid ensemble machine learning compared with other ensemble machine learners from the literature. The results from the similarity of our proposed method and the

extra boost indicate how the voting method takes advantage of the mode value of the individual base classifier in deciding the final results of the model.

Our proposed hybrid architecture can also be applied in predicting the departure or arrival of a train, bus or ship, among others. Our model treats arrival flight delays but can also be applied to study departure delays. In this case, the transportation society can replace the airline or airport with the port or station and trip distance by the flight distance. The limitation of this research that can be subject to further research is that only the US flight information was applied to predict arrival flight delay. Another limitation is that we did not consider meteorological data.

The flight delay prediction using real-time flight datasets may be considered with other machine learning algorithms in the future. Moreover, weather influences flight delays and can be the focus. This current prediction model does not add exact weather-related features but does not make weather unimportant. Contrarily, it will be significant to study the influence of weather on flight delays. We will be focusing on establishing reasonable features to measure the high impact of weather on flight delays using machine learning to analyse the relationship between flight delay and weather. Furthermore, the deep learning model will be an interesting architecture to investigate. Finally, we would study and fulfilled all the needs and lacks in this research.

Acknowledgements and Funding

The Petroleum Trust Development Fund (PTDF) Nigeria partially funded this work through grant number PTDF/ED/OSS/PHD/DBB/1558/19 for the first author's PhD studies. We also thank the UKRI for the Covid-19 recovery grant under the budget code SA077N.

Reference

[1] Bisandu, D. B., Moulitsas, I., and Filippone, S., "Social ski driver conditional autoregressive-based deep learning classifier for flight delay prediction," *Neural Comput. Appl.*, vol. 34, no. 11, pp. 8777–8802, 2022, doi: 10.1007/s00521-022-06898-y.

[2] Bisandu, D. B., Homaid, M. S., Moulitsas, I., and Filippone, S., "A deep feedforward neural network and shallow architectures effectiveness comparison: Flight delays classification perspective," in *The 5th International Conference on Advances in Artificial Intelligence (ICAAI 2021) in QAHE at Northumbria University London Campus, UK*, 2021, pp. 1–10, doi: https://doi.org/10.1145/3505711.3505712.

[3] Yi, J., Zhang, H., Liu, H., Zhong, G., and Li, G., "Flight Delay Classification Prediction Based on Stacking Algorithm," *J. Adv. Transp.*, vol. 2021, pp.1-10. doi: 10.1155/2021/4292778.

[4] Alharbi B., and Prince, M., "A hybrid artificial intelligence approach to predict flight delay," *Int. J. Eng. Res. Technol.*, vol. 13, no. 4, pp. 814–822, 2020.

[5] Perrell, E., "Embry-Riddle Aeronautical University," *AIAA Sp. 2013 Conf. Expo.*, 2013.

[6] Kim, Y. J., Choi, S., Briceno, S., and Mavris, D., "A deep learning approach to flight delay prediction," in *AIAA/IEEE Digital Avionics Systems Conference - Proceedings*, 2016, vol. 2016-Decem, pp. 1–6, doi: 10.1109/DASC.2016.7778092.

[7] Al-Tabbakh, S. M., Mohamed, H. M., and El, Z. H., "Machine learning techniques for analysis of Egyptian flight delay," *Int. J. Data Min. Knowl. Manag. Process*, vol. 8, no. 3, pp. 01–14, 2018, doi: 10.5121/ijdkp.2018.8301.

[8] J. Chen and M. Li, "Chained predictions of flight delay using machine learning," in *AIAA Scitech 2019 Forum*, 2019, no. January, pp. 1–25, doi: 10.2514/6.2019-1661.

[9] Alla, H., Moumoun, L., and Balouki, Y., "A Multilayer Perceptron Neural Network with Selective-Data Training for Flight Arrival Delay Prediction," *Sci. Program.*, vol. 2021, 2021, doi: 10.1155/2021/5558918.

[10] de Silva, B. M., Callaham, J., Jonker, J., Goebel, N., Klemisch, J., McDonald, D., ... and Aravkin, A. Y., "Hybrid Learning Approach to Sensor Fault Detection with Flight Test Data," *AIAA J.*, vol. 59, no. 9, pp. 3490–3503, 2021, doi: 10.2514/1.j059943.

[11] Yan, Z., Yang, H., Li, F., and Lin, Y., "A deep learning approach for short-term airport traffic flow prediction," *Aerospace*, vol. 9, no. 1, pp. 1–15, 2022, doi: 10.3390/aerospace9010011.

[12] Liu, F., Sun, J., Liu, M., Yang, J., and Gui, G., "Generalised Flight Delay Prediction Method Using Gradient Boosting Decision Tree," *IEEE Veh. Technol. Conf.*, vol. 2020-May, no. May 2019, 2020, doi: 10.1109/VTC2020-Spring48590.2020.9129110.

[13] Sharma Priyadarshini, M., Venturi, S., and Panesi, M., "Application of DeepOnet to model inelastic scattering probabilities in air mixtures," pp. 1–13, 2021, doi: 10.2514/6.2021-3144.

[14] Guo, Z., Hao, M., Yu, B., and Yao, B., "Detecting delay propagation in regional air transport systems using convergent cross mapping and complex network theory," *Transp. Res. Part E Logist. Transp. Rev.*, vol. 157, no. January 2021, p. 102585, 2022, doi: 10.1016/j.tre.2021.102585.

[15] Yu, Y., Chen, H., Yuan, L., and Zhang, B., "Flight delay classification warning based on evolutionary under-sampling bagging ensemble learning," vol. 1205821, no. December 2021, p. 100, 2021, doi:

10.1117/12.2619725.
[16] Sternberg, A., Soares, J., Carvalho, D., and Ogasawara, E., "A Review on Flight Delay Prediction," pp. 1–21, 2017, doi: 10.1080/01441647.2020.1861123.
[17] Balakrishna, P., Ganesan, R., and Sherry, L., "Accuracy of reinforcement learning algorithms for predicting aircraft taxi-out times: A case-study of Tampa Bay departures," *Transp. Res. Part C Emerg. Technol.*, vol. 18, no. 6, pp. 950–962, 2010, doi: 10.1016/j.trc.2010.03.003.
[18] Rodríguez-Sanz, Á., Comendador, F. G., Valdés, R. A., Pérez-Castán, J., Montes, R. B., and Serrano, S. C., "Assessment of airport arrival congestion and delay: Prediction and reliability," *Transp. Res. Part C Emerg. Technol.*, vol. 98, no. November 2018, pp. 255–283, 2019, doi: 10.1016/j.trc.2018.11.015.
[19] Cheng, C., Xu, W., and Wang, J., "A comparison of ensemble methods in financial market prediction," *Proc. 2012 5th Int. Jt. Conf. Comput. Sci. Optim. CSO 2012*, pp. 755–759, 2012, doi: 10.1109/CSO.2012.171.
[20] Henriques R., and Feiteira, I., "Predictive modelling: Flight delays and associated factors, Hartsfield-Jackson Atlanta international airport," *Procedia Comput. Sci.*, vol. 138, pp. 638–645, 2018, doi: 10.1016/j.procs.2018.10.085.
[21] Choi, S., Kim, Y. J., Briceno, S., and Mavris, D., "Cost-sensitive prediction of airline delays using machine learning," *AIAA/IEEE Digit. Avion. Syst. Conf. - Proc.*, vol. 2017-Septe, 2017, doi: 10.1109/DASC.2017.8102035.
[22] Dou, X., "Flight Arrival Delay Prediction And Analysis Using Ensemble Learning," no. Itnec, pp. 836–840, 2020, doi: 10.1109/itnec48623.2020.9084929.
[23] Etxebarria, I. E., Ciruelos, C. C., Fleurquin, P. F., Arranz, A. A., Campanelli, B. C., Eguíluz, V. M. E., and Ramasco, J. J. R., "Comparing the modeling of delay propagation in the US and European air traffic networks," *J. Air Transp. Manag.*, vol. 56, no. Part A, pp. 12–18, 2016, doi: 10.1016/j.jairtraman.2016.03.017.
[24] Hendrickx, R., Zoutendijk, M., Mitici, M., and Schafer, J., "Considering Airport Planners' Preferences and Imbalanced Datasets when Predicting Flight Delays and Cancellations," pp. 1–10, 2021, doi: 10.1109/dasc52595.2021.9594367.
[25] Mueller E. R., and Chatterji, G. B., "Analysis of aircraft arrival and departure delay characteristics," in *AIAA's Aircraft Technology, Integration, and Operations (ATIO) 2002 Technical Forum*, 2002, no. October, pp. 1–14, doi: 10.2514/6.2002-5866.
[26] Wu, Q., Hu, M., Ma, X., Wang, Y., Cong, W., and Delahaye, D., "Modeling Flight Delay Propagation in Airport and Airspace Network," *IEEE Conf. Intell. Transp. Syst. Proceedings, ITSC*, vol. 2018-Novem, pp. 3556–3561, 2018, doi: 10.1109/ITSC.2018.8569657.
[27] Tu, Y., Ball, M. O., and Jank, W. S., "Estimating flight departure delay distributions - A statistical approach with long-term trend and short-term pattern," *J. Am. Stat. Assoc.*, vol. 103, no. 481, pp. 112–125, 2008, doi: 10.1198/016214507000000257.
[28] Wu F., and Wu, F., "DeepETA: A Spatial-Temporal Sequential Neural Network Model for Estimating Time of Arrival in Package Delivery System," *Proc. AAAI Conf. Artif. Intell.*, vol. 33, pp. 774–781, 2019, doi: 10.1609/aaai.v33i01.3301774.
[29] Abdella, J. A., Zaki, N., Shuaib, K., and Khan, F., "Airline ticket price and demand prediction: A survey," *J. King Saud Univ. - Comput. Inf. Sci.*, no. xxxx, 2019, doi: 10.1016/j.jksuci.2019.02.001.
[30] Pyrgiotis, N., Malone, K. M., and Odoni, A., "Modelling delay propagation within an airport network," *Transp. Res. Part C Emerg. Technol.*, vol. 27, pp. 60–75, 2013, doi: 10.1016/j.trc.2011.05.017.
[31] Xu, N., Sherry, L., and Laskey, K. B., "Multifactor model for predicting delays at US airports," *Transp. Res. Rec.*, no. 2052, pp. 62–71, 2008, doi: 10.3141/2052-08.
[32] Belcastro, L., Marozzo, F., Talia, D., and Trunfio, P., "Using scalable data mining for predicting flight delays," ACM Trans. Intell. Syst. Technol., vol. 8, no. 1, 2016, doi: 10.1145/2888402.
[33] Horiguchi, Y., Baba, Y., Kashima, H., Suzuki, M., Kayahara, H., and Maeno, J., "Predicting Fuel Consumption and Flight Delays for Low-Cost Airlines," Innov. Appl. Artif. Intell. Conf., pp. 4686–4693, 2017, doi: https://doi.org/10.1609/aaai.v31i2.19095.
[34] Moretti, F., Pizzuti, S., Panzieri, S., and Annunziato, M., "Urban traffic flow forecasting through statistical and neural network bagging ensemble hybrid modeling," *Neurocomputing*, vol. 167, pp. 3–7, 2015, doi: 10.1016/j.neucom.2014.08.100.
[35] Okun O., Valentini G., and Re, M. (Eds.). *Ensembles in machine learning applications* (Vol. 373). Springer Science & Business Media. 2011. 273.
[36] Mohareb, F., Papadopoulou, O., Panagou, E., Nychas, G. J., and Bessant, C., "Ensemble-based support vector machine classifiers as an efficient tool for quality assessment of beef fillets from electronic nose data," *Anal. Methods*, vol. 8, no. 18, pp. 3711–3721, 2016, doi: 10.1039/c6ay00147e.
[37] Peng, Y., "A novel ensemble machine learning for robust microarray data classification," *Comput. Biol. Med.*, vol. 36, no. 6, pp. 553–573, 2006, doi: 10.1016/j.compbiomed.2005.04.001.

[38] Cavalcanti, P. G., Scharcanski, J., and Baranoski, G. V. G., "A two-stage approach for discriminating melanocytic skin lesions using standard cameras," *Expert Syst. Appl.*, vol. 40, no. 10, pp. 4054–4064, 2013, doi: 10.1016/j.eswa.2013.01.002.
[39] Levi, Y., Bekhor, S., and Rosenfeld, Y., "A multi-objective optimisation model for urban planning: The case of a very large floating structure," *Transp. Res. Part C Emerg. Technol.*, vol. 98, no. November 2018, pp. 85–100, 2019, doi: 10.1016/j.trc.2018.11.013.
[40] Yu, B., Guo, Z., Asian, S., Wang, H., and Chen, G., "Flight delay prediction for commercial air transport: A deep learning approach," *Transp. Res. Part E Logist. Transp. Rev.*, vol. 125, no. March, pp. 203–221, 2019, doi: 10.1016/j.tre.2019.03.013.
[41] Fernandes, N., Moro, S., Costa, C. J., and Aparício, M., "Factors influencing charter flight departure delay," *Res. Transp. Bus. Manag.*, vol. 34, no. December, p. 100413, 2019, doi: 10.1016/j.rtbm.2019.100413.
[42] Yao, S., *Flight Arrival Delay Prediction Using Gradient Boosting Classifier*, vol. 813, no. January. Springer Singapore, 2019.
[43] Mangortey, E., Puranik, T. G., Pinon-Fischer, O. J., and Mavris, D. N., "Classification, Analysis, and Prediction of the Daily Operations of Airports Using Machine Learning," no. January, pp. 1–18, 2020, doi: 10.2514/6.2020-1196.
[44] Hesam, M., Sultana, N., Millward, H., and Liu, L., "Ensemble learning activity scheduler for activity based travel demand models," *Transp. Res. Part C*, vol. 123, no. February, p. 102972, 2021, doi: 10.1016/j.trc.2021.102972.
[45] de Juniac, A., "International Air Transport Association (IATA) - Annual review 2019," *75th Annu. Gen. Meet. Seoul, June 2019*, 2019.
[46] IATA, "AI in Aviation," no. June, pp. 1–20, 2018.
[47] Industry High-level Group, "Aviation Benefits Report," p. 76, 2019.
[48] Boksberger, P., "Aviation Systems," *Aviat. Syst.*, pp. 157–170, 2011, doi: 10.1007/978-3-642-20080-9.
[49] Fleurquin, P., Ramasco, J. J., and Eguiluz, V. M., "Systemic delay propagation in the US airport network," *Sci. Rep.*, vol. 3, 2013, doi: 10.1038/srep01159.
[50] Moreno-Torres, J. G. , Saez, J. A., and Herrera, F., "Study on the impact of partition-induced dataset shift on k-fold cross-validation," *IEEE Trans. Neural Networks Learn. Syst.*, vol. 23, no. 8, pp. 1304–1312, 2012, doi: 10.1109/TNNLS.2012.2199516.
[51] Chander, S., Padmanabha, S., and Mani, J., "Jaya Spider Monkey Optimization-driven Deep Convolutional LSTM for the prediction of COVID'19," *Bio-Algorithms and Med-Systems*, vol. 16, no. 4. 2020, doi: 10.1515/bams-2020-0030.

A hybrid ensemble machine learning approach for arrival flight delay classification prediction using voting aggregation technique

Bisandu, Desmond Bala

2023-06-08

Bisandu DB, Moulitsas I. (2023) A hybrid ensemble machine learning approach for arrival flight delay classification prediction using voting aggregation technique. In: 2023 AIAA Aviation and Aeronautics Forum and Exposition (AIAA AVIATION Forum), 12-16 June 2023, San Diego, USA. Paper number AIAA 2023-4326

https://doi.org/10.2514/6.2023-4326