Hindawi
Journal of Advanced Transportation
Volume 2021, Article ID 4292778, 10 pages
https://doi.org/10.1155/2021/4292778

Research Article

Flight Delay Classification Prediction Based on Stacking Algorithm

Jia Yi,[1] Honghai Zhang,[1] Hao Liu,[2] Gang Zhong,[1] and Guiyi Li[1]

[1]*College of Civil Aviation, Nanjing University of Aeronautics&Astronautics, Nanjing 211106, China*
[2]*College of Science, Nanjing University of Aeronautics&Astronautics, Nanjing 211106, China*

Correspondence should be addressed to Honghai Zhang; zhh0913@163.com

Received 2 June 2021; Revised 19 July 2021; Accepted 11 August 2021; Published 18 August 2021

Academic Editor: Chi-Hua Chen

With the development of civil aviation, the number of flights keeps increasing and the flight delay has become a serious issue and even tends to normality. This paper aims to prove that Stacking algorithm has advantages in airport flight delay prediction, especially for the algorithm selection problem of machine learning technology. In this research, the principle of the Stacking classification algorithm is introduced, the SMOTE algorithm is selected to process imbalanced datasets, and the Boruta algorithm is utilized for feature selection. There are five supervised machine learning algorithms in the first-level learner of Stacking including KNN, Random Forest, Logistic Regression, Decision Tree, and Gaussian Naive Bayes. The second-level learner is Logistic Regression. To verify the effectiveness of the proposed method, comparative experiments are carried out based on Boston Logan International Airport flight datasets from January to December 2019. Multiple indexes are used to comprehensively evaluate the prediction results, such as Accuracy, Precision, Recall, *F*1 Score, ROC curve, and AUC Score. The results show that the Stacking algorithm not only could improve the prediction accuracy but also maintains great stability.

1. Introduction

Airports are significant nodes of air transportation. The number of airport flight delays has been on increase in recent years. Delayed flights are defined by the Federal Aviation Administration when they arrive or depart more than 15 minutes later than scheduled. In 2019, the arrival delay rate is 19.2% and the departure delay rate is 18.18% in the United States [1]. Flight delays can cause many negative effects, such as passengers' inconvenience, increased airport pressure, and airline losses [2]. Effective flight delay prediction could provide support for flight plan and emergency plan formulation, reduce the economic loss, and alleviate the negative impact. The Bureau of Transportation Statistics has recorded the nationwide flight operation data in the United States which provides valuable and reliable datasets for study flight delay issues. Meanwhile, with the development of artificial intelligence, machine learning technology has been widely used in airport flight delay prediction. Machine learning technology involves multiple disciplines, such as probability, statistics, and computer science [3]. Machine learning can break the limitations of mathematical formulas and improve the accuracy of flight delay prediction. In general, machine learning technology can be roughly divided into supervised learning, unsupervised learning, deep learning, reinforcement learning, and ensemble learning. Each of these learning methods has its characteristics. We should select the appropriate methods and algorithms to carry on research. Poorly performing algorithms not only cannot gain accurate results but also wastes computing power. Therefore, algorithm selection is an important process in machine learning technology. This paper aims to provide an applicable flight delay classification prediction method, especially for solving algorithm selection problems.

Many scholars have studied flight delay issue based on different machine learning methods. Esmaeilzadeh and Mokhtarimousavi used a support vector machine to mine the nonlinear relationship between flight delay and various features. Given the black-box nature of machine learning, the sensitivity analysis of corresponding variables and

independent variables was conducted, and weather factors, airport scene operation, demand, and other factors were comprehensively considered. This research provided a new idea for studying the flight delay causes [3]. Kalyani et al. proposed a flight arrival delay prediction classification model based on XGBoost and a flight arrival delay prediction regression model based on linear regression. As one of the most widely used algorithms in the machine learning field, linear regression has the advantages of simple principle and easy application, and XGBoost is an ensemble learning algorithm based on the Decision Tree, which can find the optimal result by constantly adjusting the hyperparameters [4]. Zhang and Ma established a flight delay prediction model based on the Catboost algorithm, and the prediction accuracy reached 0.77. The SHAP value was used to analyze the features' contribution degree [5]. Khaksar and Sheikholeslami developed a hybrid method combining the J48 Decision Tree with *K*-means to train flight datasets from the United States and Iran, respectively, and compared them with four algorithms and obtained the optimal results with the hybrid method [6].

When utilizing machine learning techniques, most scholars will use multiple machine learning algorithms to train the same datasets and come up with the optimal algorithm and the optimal predict result through the evaluation indexes comparison [7, 8]. Moreover, with the development of machine learning technology, the variety of algorithms is increasing and most scholars tend to use at least three algorithms in one research. Henriques and Feiteira presented a classification model based on Hartsfield-Jackson International Airport which utilized Decision Tree, Random Forest, and Multilayer Perceptron. The Multilayer Perceptron provided the highest accuracy [9]. Choi et al. attempted two supervised learning algorithms, Decision Tree and KNN, and two ensemble learning algorithms, Random Forest, and Adaboost, and the results showed that ensemble algorithm classifier was greater than single algorithm classifier [10]. Stefanovič et al. took Lithuania Airport flight delays datasets as the research object and selected seven machine learning algorithms including probabilistic neural network, multilayer perceptron neural network, Gradient-Boosted Tree, Decision Tree, and the Gradient-Boosted Tree obtained the optimal results [11]. The above research studies are inspirational, and most of them through the model comparison obtain one optimal model while the other models were eliminated which create a waste of computing power. In addition, flight datasets are enormous and versatile, and the stability of algorithm is significant for real world applications. However, most studies did not pay attention to the algorithm stability, especially some novel algorithms. In thie study, we build a flight delay prediction classification model based on Stacking and design the experiments to verify the stability of Stacking.

The flight delay prediction methods based on machine learning technology become mature gradually. However, one core process that is often neglected in previous studies is feature selection [12]. Features selection is an essential step in machine learning [13]. The main purpose of feature selection is to remove redundant features and improve model efficiency by calculating feature importance. Onan and Korukoglu presented a feature selection model based on the ensemble method. The experiment result shows that the proposed method not only effectively processed the complex features but also improved the classification accuracy [14]. In addition, considering weather information could effectively improve the prediction accuracy [15], but the exact weather information might not be available until few hours before the flight. Therefore, we are not considering bringing in weather features in this research temporarily. The rest of this paper is organized as follows. Section 2 elaborates the research methods and principles used in this study including the Stacking classification algorithm, the SMOTE algorithm, the Boruta algorithm, and several indexes. Section 3 describes the data sources and the data preprocessing method. Section 4 discusses comparative experiments and comprehensively evaluates the prediction results through Accuracy, Precision, Recall, *F*1 Score, ROC curve, and AUC Score. In Section 5, the conclusions and expectations of this research are discussed.

2. Methodologies

2.1. Stacking Classification Methods. Stacking methods are derived from the idea of ensemble learning based on learners' combinations [16]. Stacking learner usually contains two levels, the first-level learner consists of multiple basics learners selected for training the same datasets, and the predicted outputs will become a new dataset to be carried into the second-level learner [17]. To avoid overfitting, cross-validation can be used when the first-level learner is the training model, and we select the *k*-fold cross-validation method in this paper [18]. The main process of Stacking methods is shown in Figure 1.

The initial datasets have been divided into training dataset *Dta* and testing dataset *Dts*, and then the training dataset *Dta* has been divided into *k* subdatasets, *Dta*1, *Dta*2,. . ., *Dtak*. In the *k*-fold cross-validation method, *i* models will be trained for *k* times, each subdataset becomes a test dataset in turn, and other subdatasets are training datasets to participate in training. In each model, *k* prediction results are combined to form a new training subdataset $Tir(r = 1,2, \ldots, k)$ and Tir $(r = 1,2, \ldots, k)$ have formed a new training datasets *Nta* and brought into the second-level learner.

When K-fold cross-validation is carried out in the first-level learner, every time Model *i* trains the training dataset *Dta,* testing datasets *Dts* will be predicted as well. Therefore, *k* prediction results *Rik* which are predicted by the same testing dataset *Dts* will be obtained. When solving the regression problem, the averaging method is usually adopted to process the *k* prediction results. In the classification problem, the processing of the prediction results is shown in Figure 2.

In machine learning, the binary classification will output the probability value of positive and negative at first. The category corresponding to a higher probability value is the category of the data sample, and the sum of the probability value is 1. In Stacking classification, model *i* predicts that the

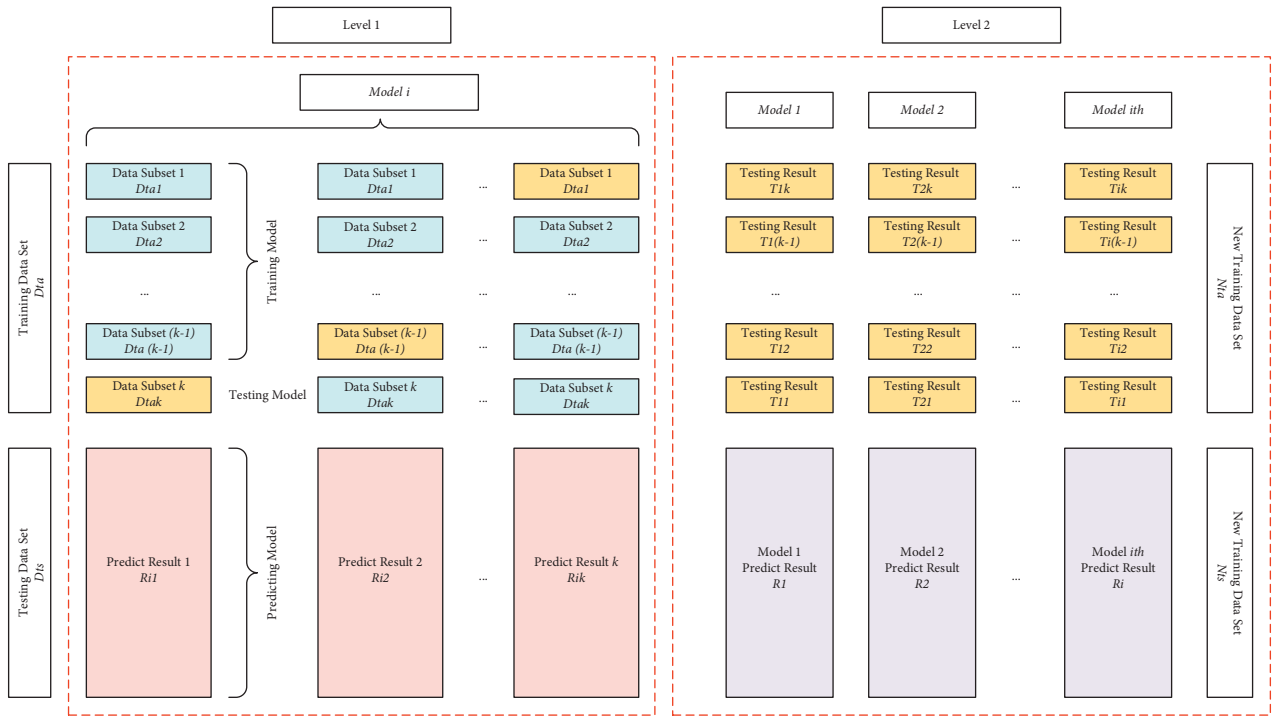

Figure 1: Stacking methods framework.

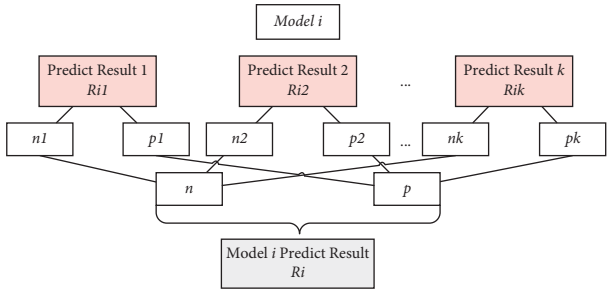

Figure 2: Stacking classification method of the second-level learner framework.

probability of the data sample belonging to positive p is $P(p) = (p1 + p2 + \cdots + pk)/k$ and the probability of the data sample belonging to negative is $P(n) = (n1 + n2 + \cdots + nk)/k$. Thus, the prediction result of Model i on testing dataset Dts, Ri ($i = 1,2, \ldots, i$), forms a new testing dataset Nts into the second-level learner. The second-level learner could choose a relatively simple algorithm and then trains the model with the new training dataset Nta and test with new testing datasets Nts.

2.2. Imbalanced Datasets Processing. Imbalanced datasets are one of the common problems in machine learning classification. This is mainly reflected in the fact that the number of samples belonging to a certain category in the datasets is far greater than that of other categories. To improve the accuracy, most classification algorithms tend to identify the minority class data samples as the majority class samples when training imbalanced datasets. Although such a classifier can achieve a certain accuracy, it does not have applicability [19]. The flight delay datasets in this paper are typical imbalanced datasets, and the data volume of on-time flights is nearly four times that of delayed flights (3.78 : 1).

Oversampling and undersampling are the commonly used techniques to deal with imbalanced datasets [20]. The main idea of these two technologies is to reconstruct the sample size. Undersampling has achieved balance by reducing most samples, while Oversampling has achieved balance by increasing the minority of samples.

In this paper, SMOTE (synthetic minority oversampling technique) algorithm is selected to process the imbalanced datasets [21]. The SMOTE algorithm is an oversampling technology based on the KNN algorithm. It improves the simple random oversampling algorithm of randomly copying a few samples to increase the sample size, which can avoid overfitting and effectively improve the generalization ability of the model. The main process of the SMOTE algorithm is as follows:

(1) The Euclidean distance is calculated from each minority sample x to the other minority sample

(2) The sampling rate is set according to the difference between the minority sample size and the majority sample size and randomly determines k nearest neighbors of sample x of a minority class

(3) Between a few samples x and x_i, according to the sampling rate set in Step (2), a new sample x_n can be calculated according to the following formula:

$$x_n = x + ran\ d(0,1) \times \left| x - x_i \right|. \quad (1)$$

2.3. Features Selection. Feature selection is one of the core contents of machine learning, which aims to eliminate redundant features, improve model accuracy, and reduce operation time. The commonly used feature selection methods include Filter, Wrapper, and Embedded [22]. The Boruta algorithm is utilized in this research to select features. Boruta is an encapsulated feature selection algorithm based on Random Forest. The importance of each feature to the dependent variable is calculated to determine whether to be retained. The main process of the Boruta algorithm is as follows:

(1) Establish shadow feature: the original features are randomly sorted to form a shadow feature matrix, and the new feature matrix is obtained by splicing the shadow feature matrix with the original feature matrix.

(2) The new feature matrix is brought in a Random Forest classifier for training, and output the importances of features v.

(3) The Z score of the original feature and shadow feature is calculated, and the calculation formula is as follows:

$$z_{\text{score}} = \frac{A_v}{S_v}, \quad (2)$$

where $A_v$represents the average value of feature importance and $S_v$represents the standard deviation of feature importance.

(4) The maximum z_{score} is searched in the shadow feature, denoted as Z_{max}.

(5) If the original feature z_{score} is greater than Z_{max}, the feature is recorded as "important." On the contrary, if the original feature z_{score} is less than Z_{max}, the feature will be marked as "unimportant" and be deleted.

(6) Steps (1) to (5) are repeated until all features have been marked.

2.4. Evaluation Indexes. In this paper, Accuracy, Precision, Recall, and $F1$ Score are calculated by output confusion matrix to evaluate the prediction results. The confusion matrix is shown in Figure 3 [23].

TP is True Positive, indicating that both the true value and the predicted value are positive, that is, the number of positive samples predicted correctly. FP is False Positive, indicating that the true value is negative, but the predicted value is positive, that is, the number of negative samples is wrongly predicted to be positive. TN is True Negative, indicating that both the true value and the predicted value are negative, that is, the number of negative samples that are correctly predicted. FN is False Negative, indicating that the true value is positive, but the predicted value is negative, that is, the number of positive samples that are wrongly predicted to be negative.

Confusion Matrix		Predicted	
		Positive	Negative
True	Positive	*TP*	*FN*
	Negative	*FP*	*TN*

Figure 3: Confusion matrix.

Accuracy is the ratio of correctly predicted samples to the total amount of samples, and its calculation formula is as follows:

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{FN} + \text{TN}} \cdot 100\%. \quad (3)$$

Accuracy is one of the most used evaluation indexes in classification. Since the flight delay data sample is the imbalanced dataset, that is, the sample size of on-time flights is much larger than delayed flights. To improve accuracy, the model tends to identify the minority samples as the majority, and the model can obtain higher accuracy, but the prediction of delayed samples is almost ineffective. Therefore, the predicted results also need to be evaluated by Precision, Recall, and $F1$ Score in the classification problem.

Precision indicates the percentage of correct predictions in the sample with a positive predicted value. The calculation formula is as follows:

$$\text{Precision} = \frac{\text{TP}}{(\text{TP} + \text{FP})} \cdot 100\%. \quad (4)$$

Recall indicates the percentage of the correct prediction in the sample with a positive true value. The calculation formula is as follows:

$$\text{Recall} = \frac{\text{TP}}{(\text{TP} + \text{FN})} \cdot 100\%. \quad (5)$$

According to the calculation formula of Precision and Recall, it can be found that when the Precision increases, the Recall will decrease, and when the Recall increases, the Precision will decrease. In this paper, the Precision focuses on how many delayed flights were successfully predicted in the total sample, while the Recall focuses on how many delayed flights were successfully predicted in all delayed flights. Moreover, the $F1$ Score, as the harmonic average of Precision and Recall, could consider both. The calculation formula is as follows:

$$F1\ \text{score} = \frac{2 \cdot \text{Precisiong} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}. \quad (6)$$

3. Data Acquisition and Preprocessing

3.1. Data Sources. In this research, we collect flight data from January to December 2019 at Logan International Airport in Boston, Massachusetts, the United States. The total number of departure flight datasets is 149,576, and the total number of arrival flight datasets is 149338. The Logan Airport is one

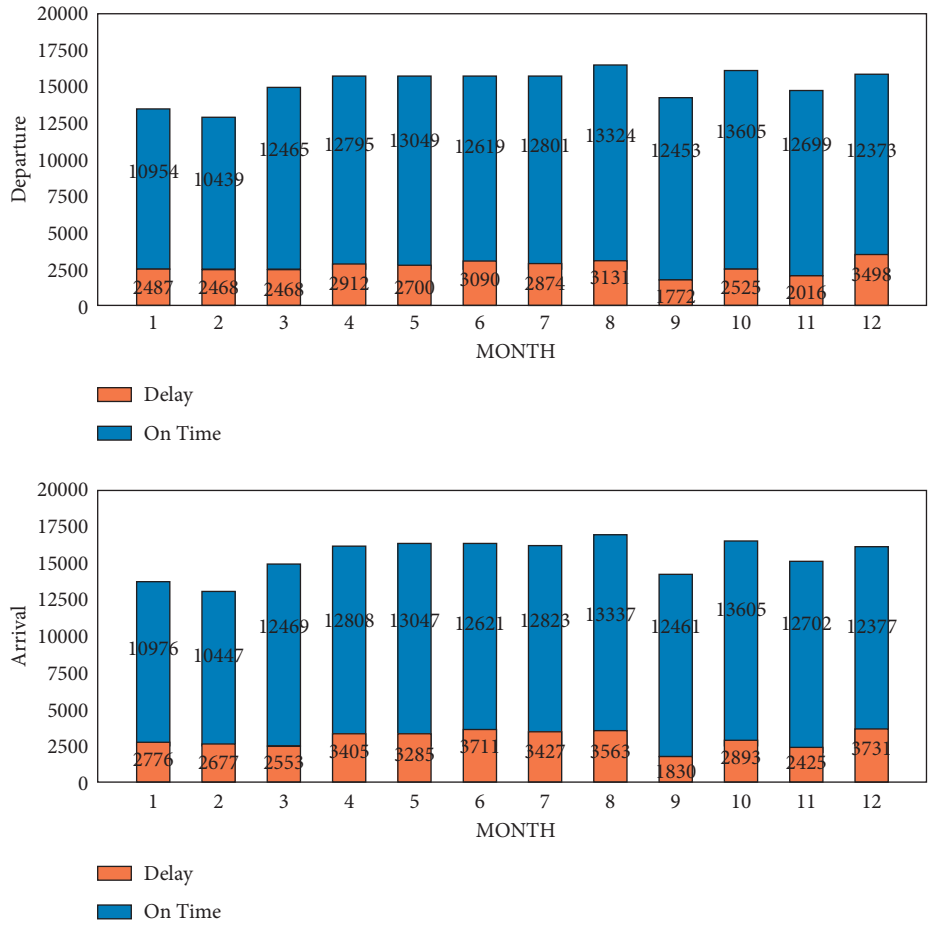

Figure 4: Monthly number of delayed flights and on-time flights.

of the busiest airports in the eastern United States, with 31, 941 flights delayed in the departure dataset and 35,941 flights delayed in the arrival dataset. The departure delay rate is 21.35%, and the arrival delay rate is 24.07%. The monthly distribution of flight delays in 2019 is shown in Figure 4.

Both datasets include 9 features, and the input features and descriptions are shown in Table 1.

3.2. Uniformization Processing. To avoid the impact of dimensionless differences among features in the dataset, the data are normalized in this paper. The aim is to adjust the mean of the data to 1 and the variance to 0. The calculation formula is as follows:

$$x' = \frac{X - X_{\text{mean}}}{X_{\max} - X_{\min}}, \tag{7}$$

where X_{mean} is the mean value, $X_{\max}$ is the maximum value, and $X_{\min}$ is the minimum value.

4. Experiment and Analysis

4.1. Features Selection Results. In this research, the Boruta algorithm is utilized to select features for the departure delay dataset and arrival dataset, respectively, and the results are shown in Figure 5. In the departure dataset, all features are marked as important. The CRS_DEP_TIME is the most important feature in the departure dataset. In the arrival dataset, 8 features are estimated as important features, and Diverted has been rejected. The departure dataset features importance is shown in Table 2, and the arrival dataset features importance is shown in Table 3.

Table 1: The input features and descriptions.

Features	Format	Description
Quarter	int64	Quarter (1–4)
Month	int64	Month (1–12)
Day_of_month	int64	Day of month (1–31)
Day_of_week	int64	Day of week (1–7)
CRS_dep_time	int64	CRS departure time (local time: hhmm)
CRS_arr_time	int64	CRS arrival time (local time: hhmm)
CRS_elapsed_time	int64	CRS elapsed time, in minutes
Distance	int64	Miles
Diverted	int64	diverted = 1, not diverted = 0

To explore the influence of features' importance on the prediction results, the following experiment has proceeded. At first, only input the most important features for training and then add one feature at a time according to the importance value until all the features are input. According to

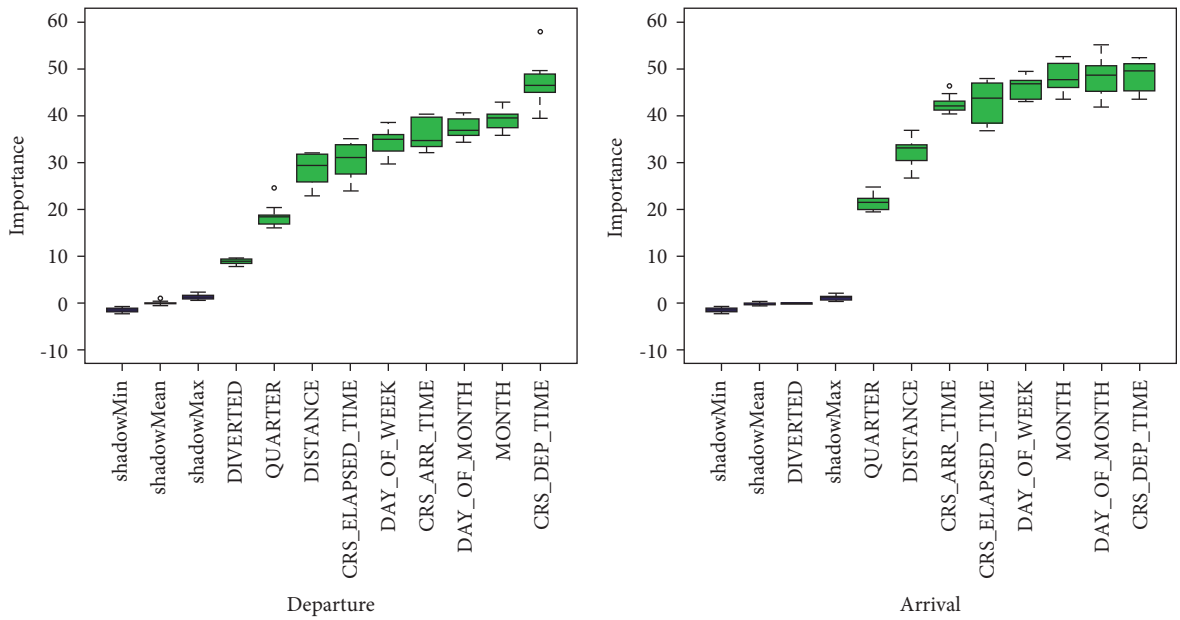

Figure 5: Features selection results: (a) departure; (b) arrival.

Table 2: Departure dataset features importance.

	meanImp	medianImp	minImp	maxImp	normHits	Decision
Quarter	18.7022	18.38765	15.96021	24.59658	1	Confirmed
Month	39.15992	39.53052	35.61664	42.88205	1	Confirmed
Day_of_month	37.13472	36.92707	34.23672	40.63482	1	Confirmed
Day_of_week	34.41572	34.80809	29.75517	38.51781	1	Confirmed
CRS_dep_time	47.17696	46.49994	39.36456	57.88962	1	Confirmed
CRS_arr_time	35.90235	34.88594	32.14023	40.42103	1	Confirmed
CRS_elapsed_time	30.42818	31.15777	24.04919	35.14513	1	Confirmed
Diverted	8.957969	9.035397	7.912347	9.665486	1	Confirmed
Distance	28.66349	29.39897	22.9603	31.81813	1	Confirmed

Table 3: Arrival dataset features importance.

	meanImp	medianImp	minImp	maxImp	normHits	Decision
Quarter	21.59	21.61953	19.64103	24.85327	1	Confirmed
Month	48.19586	47.78515	43.47335	52.73112	1	Confirmed
Day_of_month	48.35522	48.61903	41.88918	55.16763	1	Confirmed
Day_of_week	46.23737	46.70293	43.08921	49.43334	1	Confirmed
CRS_dep_time	48.68234	49.67468	43.40899	52.34451	1	Confirmed
CRS_arr_time	42.41878	42.13526	40.19638	46.36798	1	Confirmed
CRS_elapsed_time	42.78774	43.67602	36.73169	47.83175	1	Confirmed
Distance	0	0	0	0	0	Rejected
Diverted	32.28774	33.28389	26.69778	36.87651	1	Confirmed

the feature selection results, the Diverted is removed in arrival prediction model training. In this experiment, the first-level learner contains five algorithms: Decision Tree, KNN, Logistic Regression, Gaussian Naive Bayes, and Random Forest. The second-level learner is Logistic Regression. The experiment results are shown in Figure 6.

In the departure dataset, when the fifth important feature is given as input, Accuracy, Precision, Recall, and $F1$ Score exceed 0.8. When the sixth important feature is given as input, the indexes show slight decrease, but the overall trend is stable without significant increase or decrease. In other words, the last four features contributed limited to the prediction model, which was consistent with Boruta feature selection results. In the arrival dataset, when the fourth important feature is given as input, the evaluation indexes have no significant change. In the

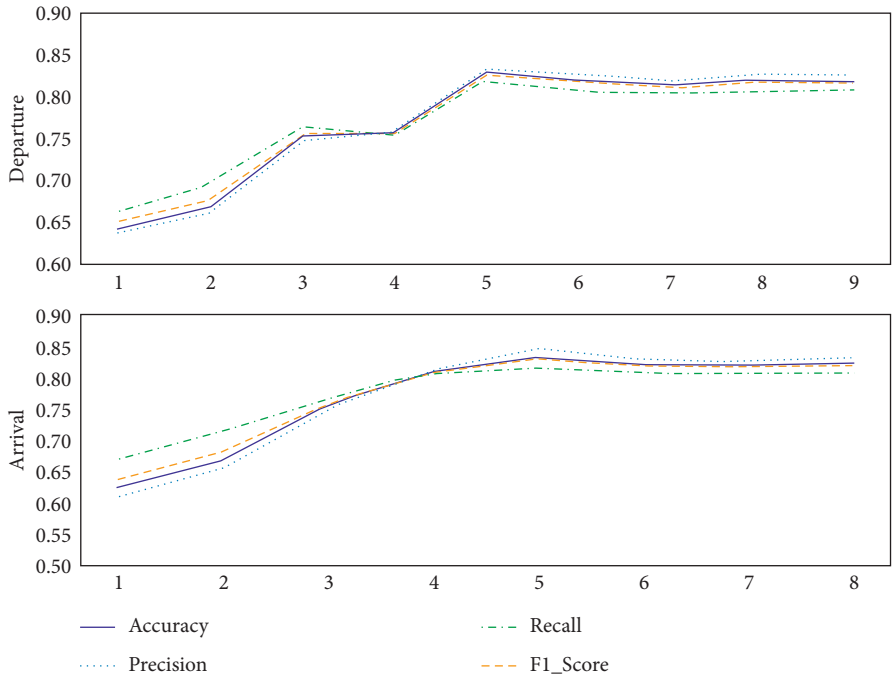

Figure 6: The prediction results with different features.

arrival dataset, when the fourth feature is given as input, the evaluation indexes exceed 0.8 and tend to be stable. It is worth mentioning that with the increase in features, Recall changes from the highest to the lowest among the four indexes, while Precision changes from the lowest to the highest.

4.2. Comparison between Algorithms. There is no "multipurpose algorithm" or "the greatest algorithm" in machine learning. It is necessary to attempt multiple algorithms. In this research, six algorithms are selected including KNN, Random Forest, Logistic Regression, Decision Tree, Gaussian Naive Bayes, and Stacking to train the same dataset, respectively. The experiment results are shown in Figure 7. In addition to Stacking, Random Forest also showed a great prediction result which four evaluation indexes all exceed 0.8. The difference among four indexes of KNN is lager than other algorithms but also has reached 0.7. Meanwhile, Gaussian Naive Bayes and Logistic Regression have relatively poor performance, and four indexes are around 0.6.

The ROC (receiver operating characteristic) curve could measure algorithm generalization ability. The AUC (area under curve) is the area under the ROC curve [24]. The closer the AUC is to 1, the better the algorithm will be. We output the ROC for each algorithm and calculate the AUC Score, and the results are shown in Figure 8. Stacking reaches 0.823 in the departure dataset and 0.821 in the arrival dataset. The result of Random Forest is similar to that of Stacking. With this result, we consider that Random Forest contributes more to Stacking compared with other algorithms. However, if we remove Random Forest from the Stacking algorithm, will the performance of Stacking decrease? In other words, if we remove the weak performance algorithm Gaussian Naive Bayes, will the performance of Stacking increase? In section 4.3, we experiment to explore the impact of strong and weak algorithms on the performance of Stacking.

4.3. First-Level Learners Analyses. In the single algorithm comparison, we find that the Random Forest has great performance, and Gaussian Naive Bayes and Logistic Regression perform poorly. In this section, one algorithm is removed, in turn, to figure out how strong or weak algorithms affect Stacking prediction results. The results are shown in Tables 4 and 5. Overall, there is no significant difference between the six groups with different first-level learners. Both in the departure dataset and arrival dataset, the four evaluation indexes are similar among the six scenarios, only the Recall and *F*1 Score of the third scenario decrease below 0.8. The overall accuracy is shown in Figure 9. The prediction accuracy is around 0.8 which is close to the result of Stacking. It can be concluded that the Stacking algorithm not only could ensure the prediction accuracy but also maintains great stability. Random Forest has a strong performance, but when we remove Random Forest form the first-level learner, the model still acquires great predict results. As we mentioned before, there is no "multipurpose algorithm" or "the greatest algorithm" in machine learning. Therefore, the Stacking algorithm could be a great solution to deal with algorithm selection, especially the enormous and complex datasets like flight datasets.

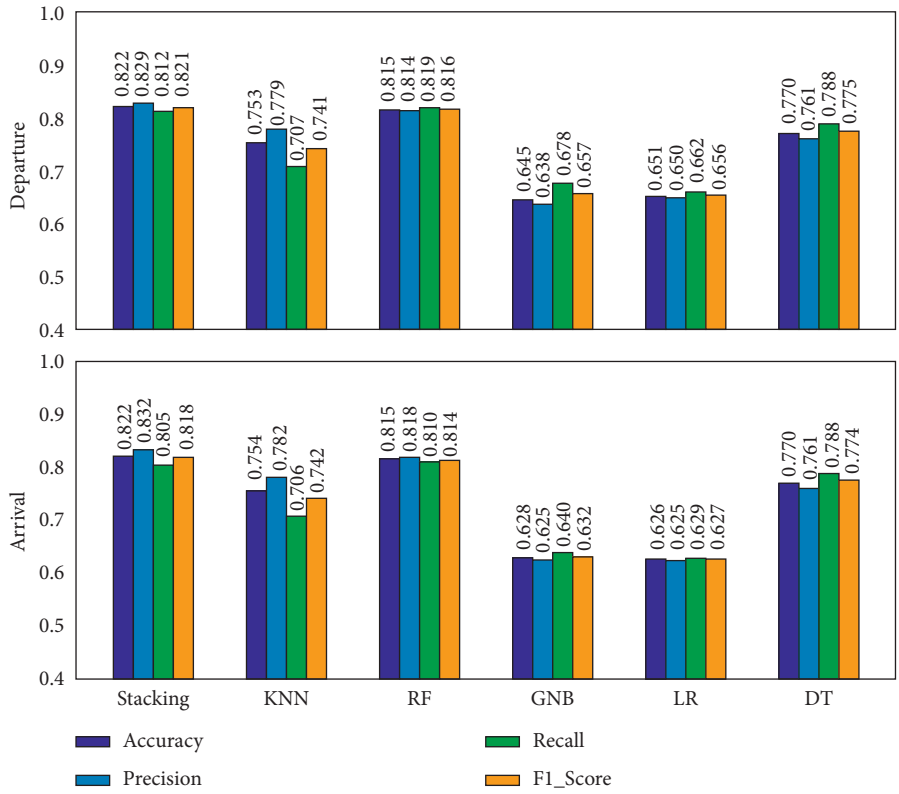

Figure 7: The prediction results of different algorithms.

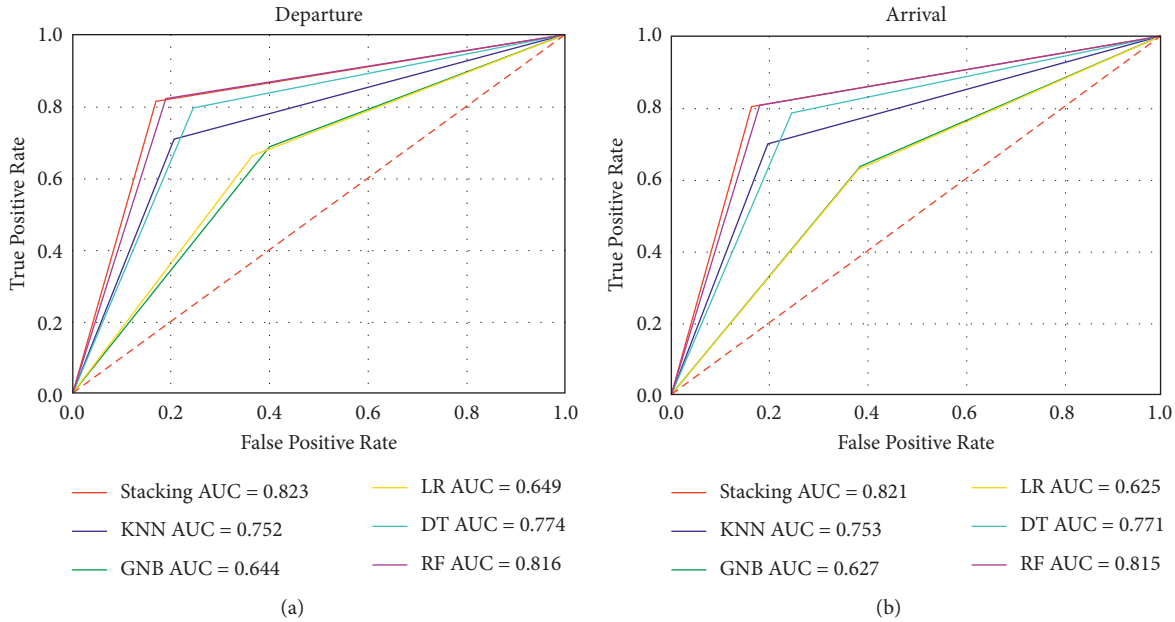

Figure 8: Receiver operating characteristic curve: (a) departure; (b) arrival.

Table 4: The departure prediction results of different first-level learners.

Departure	First-level learner	Accuracy	Precision	Recall	*F*1 Score
1	GNB, RF, KNN, LR, DT	0.822	0.830	0.812	0.821
2	RF, KNN, LR, DT	0.821	0.8277	0.812	0.820
3	GNB, KNN, LR, DT	0.800	0.805	0.784	0.794
4	GNB, RF, LR, DT	0.819	0.823	0.812	0.817
5	GNB, RF, KNN, DT	0.822	0.828	0.811	0.819
6	GNB, RF, KNN, LR	0.82	0.827	0.811	0.819

Table 5: The arrival departure prediction results of different first-level learners.

Arrival	First-level learner	Accuracy	Precision	Recall	*F1* Score
1	GNB, RF, KNN, LR, DT	0.822	0.832	0.808	0.82
2	RF, KNN, LR, DT	0.82	0.827	0.806	0.816
3	GNB, KNN, LR, DT	0.80	0.811	0.78	0.793
4	GNB, RF, LR, DT	0.818	0.824	0.804	0.814
5	GNB, RF, KNN, DT	0.82	0.828	0.805	0.816
6	GNB, RF, KNN, LR	0.818	0.825	0.804	0.814

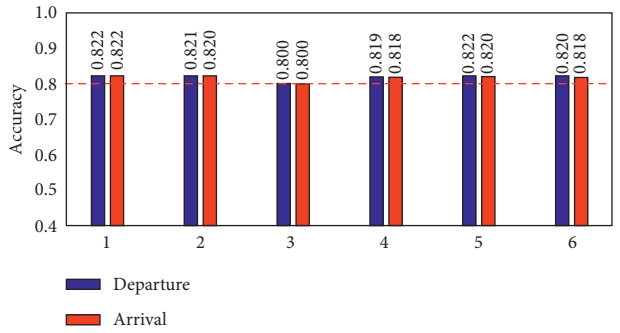

Figure 9: The accuracy of different first-level learners.

5. Conclusion

In this research, we propose a flight delay prediction classification method based on the Stacking algorithm. The SMOTE algorithm is introduced to process imbalanced datasets used, and the Boruta algorithm is utilized to select input features. The Logan International Airport flight data in 2019 are collected to carry out comparative experiments, and the Accuracy, Precision, Recall, and *F*1 Score are above 0.8. The main contributions are as follows:

(1) The Boruta algorithm is used to select features. Features selection is an essential process when utilizing machine learning technology. According to section 4.1, the comparison experimental results are consistent with the Boruta algorithm feature selection results, which verify the effectiveness of the Boruta algorithm. 9 feature importances are obtained based on the Random Forest classifier, and the experiments are designed to input different features into the model in the order of their importance value. In the departure dataset, all features have been confirmed while Diverted has been rejected in the arrival dataset.

(2) A flight delay prediction classification method based on Stacking is proposed in this study. The first-level learner includes KNN, Random Forest, Logistic Regression, Decision Tree, and Gaussian Naive Bayes, and the second-level learner utilizes Logistic Regression. To distinguish the contribution of five first-level learners, the same dataset that has been trained based on these five first-level learners separately. The result shows that Random Forest has the best performance which is similar to Stacking.

(3) The main aim of this study is to explore the stability of the Stacking algorithm. Stacking is a combination of different algorithms with different performances. In section 4.3, we design an experiment to verify how strong or weak learners affect the Stacking performance. The experiment result shows that whether strong learners or weak learners are removed, the overall accuracy of the Stacking has no obvious difference. Therefore, we believe that Stacking provides a reliable solution for algorithm selection in machine learning applications, especially the enormous and complex datasets like flight datasets.

In future research, other machine learning technologies can be utilized to study flight delay prediction. Moreover, it can also pay close attention to weather influence on a flight delay. In this research, we does not add exact weather-related features in the prediction model but that does not mean weather influence is unimportant. On the contrary, we believe that studying the influence of weather on flight delays is a significant and complex issue. We will focus more on establishing reasonable features to measure the impact of weather on flight delays, especially for high-impact weather, and use machine learning correlation analysis technology to explore the relatedness between weather and flight delay.

Data Availability

The flight dataset used in this paper is from the Bureau of Transportation Statistics website (https://www.transtats.bts.gov/homepage.asp).

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by the National Key R&D Program of China (No. 2018YFE0208700) and the National Natural Science Foundation of China (No. 52002177).

References

[1] Bureau of Transportation Statistics, "Bureau of Transportation Statistics,".

[2] M. Ball, C. Barnhart, M. Dresner et al., "Total delay impact study," 2010.

[3] E. Esmaeilzadeh and S. Mokhtarimousavi, "Machine learning approach for flight departure delay prediction and analysis," *Transportation Research Record: Journal of the Transportation Research Board*, vol. 2674, no. 8, pp. 145–159, 2020.

[4] N. L. Kalyani, G. Jeshmitha, U. Bindu Sri Sai, M. Samanvitha, J. Mahesh, and B. V. Kiranmayee, "Machine learning model - based prediction of flight delay," in *Proceedings of the 2020 Fourth International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC)*, Palladam, India, November 2020.

[5] B. Zhang and D. Ma, "Flight delay prediciton at an airport using maching learning," in *Proceedings of the 2020 5th International Conference on Electromechanical Control Technology and Transportation*, Nanchang, China, May 2020.

[6] H. Khaksar and A. Sheikholeslami, "Airline delay prediction by machine learning algorithms," *Scientia Iranica*, vol. 1, p. 12, 2017.

[7] G. Rebala, A. Ravi, and S. Churiwala, *An Introduction to Machine Learning*, Springer International Publishing, New York, NY, USA, 2019.

[8] A. Onan, "On the performance of ensemble learning for automated diagnosis of breast cancer," *Advances in Intelligent Systems and Computing*, vol. 347, pp. 119–129, 2015.

[9] R. Henriques and I. Feiteira, "Predictive modelling: flight delays and associated factors, hartsfield-jackson atlanta international airport," *Procedia Computer Science*, vol. 138, pp. 638–645, 2018.

[10] S. Choi, Y. J. Kim, B. Simon, and D. Mavris, "Prediction of weather-induced airline delays based on machine learning algorithms," in *Proceedings of the 2016 IEEE/AIAA 35th Digital Avionics Systems Conference (DASC)*, Sacramento, CA, USA, December 2016.

[11] P. Stefanovič, R. Štrimaitis, and O. Kurasova, "Prediction of flight time deviation for Lithuanian airports using supervised machine learning model," *Computational Intelligence and Neuroscience*, vol. 2020, Article ID 8878681, 10 pages, 2020.

[12] G. Gui, F. Liu, J. Sun, J. Yang, Z. Zhou, and D. Zhao, "Flight delay prediction based on aviation big data and machine learning," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 1, pp. 140–150, 2020.

[13] N. Chakrabarty, "A data mining approach to flight arrival delay prediction for american airlines," 2019, https://arxiv.org/abs/1903.06740.

[14] A. Onan and S. Korukoglu, "A feature selection model based on genetic rank aggregation for text sentiment classification," *Journal of Information Science*, vol. 99, pp. 1103–1107, 2015.

[15] Y. J. Kim, C. Sun, S. Briceno, and D. Mavris, "A deep learning approach to flight delay prediction," in *Proceedings of the Digital Avionics Systems Conference 2016*, Sacramento, CA, USA, December 2016.

[16] Z. H. Zhou, *Ensemble Methods: Foundations and Algorithms*, Taylor & Francis, Oxfordshire, UK, 2012.

[17] G. Zhong, T. Yin, L. Li, J. Zhang, H. Zhang, and B. Ran, "IEEE intelligent transportation systems magazine," *IEEE Intelligent Transportation Systems Magazine*, vol. 99, 2020.

[18] J. D. Rodriguez, A. Perez, and J. A. Lozano, "Sensitivity analysis of *K*-fold cross validation in prediction error estimation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 3, pp. 569–575, 2010.

[19] H. Patel, D. S. Rajput, G. T. Reddy, C. Iwendi, and O. Jo, "A review on classification of imbalanced data for wireless sensor networks," *International Journal of Distributed Sensor Networks*, vol. 16, no. 4, Article ID 812444408, 2020.

[20] A. Behzad Mirzaei, A. B. Bahareh Nikpour, and A. Nezamabadi Pour, "A clustering and density-based hybrid approach for imbalanced data classification," *Expert Systems with Applications*, vol. 164, 2020.

[21] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: synthetic minority over-sampling technique," *Journal of Artificial Intelligence Research*, vol. 16, no. 1, pp. 321–357, 2002.

[22] Q. Al-Tashi, H. Md Rais, S. Mirjalili, and H. Alhussian, *A Review of Grey Wolf Optimizer-Based Feature Selection Methods for Classification*, UTP Universiti Teknologi PETRONAS, Seri Iskandar, Malaysia, 2020.

[23] S. A. Alvarez, *An Exact Analytical Relation Among Recall, Precision, and Classification Accuracy in Information Retrieval*, Boston College, Newton, MA, USA, 2002.

[24] T. Fawcett, "An introduction to ROC analysis," *Pattern Recognition Letters*, vol. 27, no. 8, pp. 861–874, 2005.