Kim and Park *Journal of Big Data* (2024) 11:11
https://doi.org/10.1186/s40537-023-00867-5

Journal of Big Data

RESEARCH Open Access

Prediction of flight departure delays caused by weather conditions adopting data-driven approaches

Seongeun Kim[1,2] and Eunil Park[3,4*]

*Correspondence: eunilpark@skku.edu

[1] Department of Semiconductor and Display Engineering, Sungkyunkwan University, Seoul 03063, Republic of Korea
[2] Samsung Electronics, Gyeonggi, Republic of Korea
[3] Department of Interaction Science, Sungkyunkwan University, 25-2 Sungkyunkwan-ro, Jongno-gu, Seoul 03063, Republic of Korea
[4] Teach Company, 25-2 Sungkyunkwan-ro, Jongno-gu, Seoul 03063, Republic of Korea

Abstract

In this study, we utilize data-driven approaches to predict flight departure delays. The growing demand for air travel is outpacing the capacity and infrastructure available to support it. In addition, abnormal weather patterns caused by climate change contribute to the frequent occurrence of flight delays. In light of the extensive network of international flights covering vast distances across continents and oceans, the importance of forecasting flight delays over extended time periods becomes increasingly evident. Existing research has predominantly concentrated on short-term predictions, prompting our study to specifically address this aspect. We collected datasets spanning over 10 years from three different airports such as ICN airport in South Korea, JFK and MDW airport in the United States, capturing flight information at six different time intervals (2, 4, 8, 16, 24, and 48 h) prior to flight departure. The datasets comprise 1,569,879 instances for ICN, 773,347 for JFK, and 404,507 for MDW, respectively. We employed a range of machine learning and deep learning approaches, including Decision Tree, Random Forest, Support Vector Machine, K-nearest neighbors, Logistic Regression, Extreme Gradient Boosting, and Long Short-Term Memory, to predict flight delays. Our models achieved accuracy rates of 0.749 for ICN airport, 0.852 for JFK airport, and 0.785 for MDW airport in 2-h predictions. Furthermore, for 48-h predictions, our models achieved accuracy rates of 0.748 for ICN airport, 0.846 for JFK airport, and 0.772 for MDW airport based on our experimental results. Consequently, we have successfully validated the accuracy of flight delay predictions for longer time frames. The implications and future research directions derived from these findings are also discussed.

Keywords: Flight delay, Delay prediction weather, Machine learning, LSTM

Introduction

With the increasing demand for air travel, the number of air passengers has significantly increased. The global air passenger transport market doubles every 15 years [1]. For example, as of February 2023, the revenue passenger kilometer in Asia Pacific and North America has increased by 105.4% and 25.1% relative to that in 2022, respectively. Despite a temporary decline in passenger traffic during the Covid-19 pandemic, the number of air passengers has steadily increased over the past few decades [2].

Table 1 Number of eligible passengers for compensation versus the number of total passengers

Year	Number of passenger in Europe (millions)	Rate of compensation (%)	Number of eligible passengers (millions)	Evolution (2017/2016) (%)
2017	1020	1.58	16	14.20
2016	970	1.46	14	15.60
2015	918	1.33	12	16.10

Table 2 Different types of delays

Delay type	Portion (%)	Mean and standard deviation of delay time (min)
CarrierDelay	35.30	42.08 (64.98)
WeatherDelay	**3.86**	**69.81 (100.79)**
NASDelay	38.18	31.40 (41.93)
SecurityDelay	0.40	30.40 (40.68)
LateAircraftDelay	22.27	51.27 (58.03)

Bold valuesindicate the greatest results

Meeting the increasing demand for air travel and ensuring efficient supply chain operations require the development of aviation infrastructure. This includes expanding airport facilities, updating airline fleets, and implementing effective air schedule management. Addressing these issues is crucial to provide a seamless and reliable travel experience for passengers. However, a significant challenge in delivering satisfactory services is the frequent occurrence of unexpected flight delays and cancellations [3].

According to Tileagă and Oprisan [4], the number of compensation cases due to delayed flight schedules is increasing steadily. Table 1 shows that the number of compensation recipients for air delays and cancellations is steeply increasing every year. Flight delays have significant economic consequences for both airlines and passengers, rendering it a notable issue within the aviation industry.

Table 2 show the types and proportion of delays from 2010 to 2021 at the John F. Kennedy International Airport (JFK). It reveals that weather-related delays account for a small proportion of delays (3.86%). However, weather-related delays were longer than other types of delays, with an average delay time of 69.81 min and a standard deviation of 100.79 min [5].

The frequency of abnormal weather phenomena that are known to contribute to an increase in flight delays [6] is on the rise worldwide. In addition, the regional climate determined by geographical location plays a significant role in flight operations [7]. For example, in South Korea, the total rainfall period is concentrated from July to September each year, with approximately 42.5% rainfall in July, 27.4% in August, and 12.8% in September. In addition, the region is directly affected by typhoons at the end of August through early September every year.

While previous studies on flight delay prediction have often incorporated weather information [8–10], the majority of these studies have centered around predicting delays within relatively short timeframes, typically within thresholds of 15 min or up to 4 h, primarily tailored to airline services. However, the unique context of international flights

covering vast distances across continents and oceans, with flight durations spanning from as little as 10 h to as long as 20 h, underscores the necessity for delay prediction over more extended timeframes.

Therefore, this study aims to predict flight delays over more extended timeframes (2 to 48 h) based on weather data. We focus on three well-known international airports: Incheon International Airport in South Korea (ICN), John F. Kennedy International Airport (JFK), and Chicago Midway International Airport (MDW) in the United States. In addition, we use weather information from the meteorological agencies located at each airport. "Background and related work" section reviews previous research in this area, whereas "Methodology" section presents the machine learning and deep learning models along with the evaluation methods utilized in the study. The experimental procedures and the comparison of the results across the models are presented in "Implementation and result" section. "Discussion and concluding remarks" section concludes this paper by presenting the interpretation of the results, noteworthy findings, limitations, and suggestions for future research.

Background and related work

Several studies have been conducted to forecast flight departure delays using various statistical methods, machine learning, and deep learning techniques. Table 3 provides a summary of prior flight delay detection research based on machine learning and neural network approaches.

Researchers [9, 11, 12] have utilized Bayesian modeling, clustering, classification, and regression with diverse datasets from different regions. The time span of the data varied, ranging from 1 month to 5 years, and the airports under investigation differed as well. Khaksar and Sheikholeslami [9] identified parameters that enable effective estimation of delays. They used Bayesian modeling, decision tree, cluster classification, random forest, and hybrid methods. They used 2,825,647 data for US airlines and 15,428 data for Iranian airlines. They realized an accuracy of approximately 70%.

Al-Tabbakh et al. [11] analyzed the flight delay patterns using four decision tree classifiers, including Decisionstump, J48, Random Forest, and REPTree. They utilized 512 data from a brief duration of 1 month, i.e., January 2018. The findings revealed that among the classifiers evaluated for the Egypt Airline dataset, REPTree attained the highest accuracy score of 80.3%.

Ye et al. and Atlioğlu et al. [12, 13] conducted flight delay prediction via supervised learning methods, whereas [12] employed multiple linear regression, a support vector machine, extremely randomized trees, and LightGBM. They used 105,993 data and reported the highest accuracy of 86.53%.

Atlioğlu et al. [13] studied 11 machine learning models using data obtained following feature selection and transformation. They used 8086 data and achieved F1-scores of approximately 81%.

Certain researchers predict airline delay using neural networks and hybrid models [8, 10, 14]. Kim et al. [8] investigated the effectiveness of deep learning models in predicting air traffic delays. Daily sequences of departure and arrival flight delays for individual airports were modeled using the long short-term memory (LSTM) and recurrent neural network (RNN) architecture. The accuracy of RNN improves with deeper architectures,

Table 3 Summary of prior prediction of flight delay

	Sources	Method	Datasets	Data period	Delay time (min)	Results
Machine learning	Khaksar and Sheikholeslami [9]	Bayesian modeling, decision tree, cluster classification, random forest, and hybrid method	US and Iranian airline	US: 6 months, Iran: 16 months	0–15, 15–60, 60+	Accuracy more than 70%
	Al-Tabbakh et al. [11]	Decision tree, random forest, and REPTree	Egypt airline	Jan 2018 (1 month)	–	Accuracy around 80.3%
	Ye et al. [12]	Multiple linear regression, support vector machine, extremely randomized trees, and LightGBM	Nanjing Lukou airline	Mar 1st 2017 to Feb 28th 2018	15+	Accuracy of 86.53%
	Atlioglu et al. [13]	11 machine learning models. CART, KNN, GBM, XGB, and LGBM	Dammam King Fahd International Airport	Jan 1st 2017 to Dec 9th 2019	15+	Accuracy around 82%
Neural network	Kim et al. [8]	LSTM, RNN	ATL, LAX, ORD, DFW, DEN, JFK, SFO, CLT, LAS, PHX	Jan 2010–Aug 2015	15+, 30+	Accuracy of 90.95%
	Qu et al. [10]	CBAM-CondenseNet and SimAM-CNN-MLSTM	The Civil Aviation Administration of the China East China Regional Administration (ECRA)	Mar 2018–May 2019	15–60, 60–120, 120–240, 240+	Accuracy of 89.8%, 91.36%
	Yazdi et al. [14]	Stack denoising autoencoder- levenberg marquart model, SAE-LM, SDA	The Bureau of Transportation Statistics of United State Department of Transportation	For 5 years	15+	Accuracy of 96%, 86%, 89%

exhibiting the highest performance with an accuracy of 90.95% on the Atlanta air traffic data.

Qu et al. [10] analyzed and predicted flight delays using a convolutional neural network (CNN) and RNN models that are well-suited for classification problems in the field of deep learning. They improved the CondenseNet network by incorporating CBAM modules within the CNN-based CondenseNet algorithm to develop CBAMCondenseNet. Additionally, they constructed a CNN-MLSTM network based on the CNN model and injected the SimAM module to enhance the attention of flight chain data. They used 36,287 data of China and achieved the highest accuracy score of 91.36%.

Yazdi et al. [14] designed the proposed model to output optimized results by incorporating a technique based on stack denoising autoencoder to account for the noisy flight delay data. They constructed SAE-LM based on an autoencoder and LM algorithm. The stacked denoising autoencoder is based on only denoising autoencoder. They utilized a comprehensive dataset spanning 5 years of US flight operations, comprising a total of 3,601,679 data points. The results demonstrated that the proposed model exhibited enhanced accuracy compared with the RNN model, highlighting its effectiveness in predicting flight delays. While numerous researchers have utilized state-of-the-art machine learning and deep learning techniques to study weather-related takeoff delays from various angles, the majority of studies have focused on predicting delays within a time criterion of approximately 15 min. There has been limited exploration and prediction of flight delays exceeding 2 h.

By employing established research methodologies, it is feasible to aggregate the outcomes of short-term predictions to generate long-term forecasts. Nevertheless, it's vital to recognize that repeated predictions may introduce inaccuracies. When assessing the practical utility of such models, the ability to predict aviation delays over extended time intervals based on input data widens the scope of possibilities for long-haul flights and diverse flight schedules. This expanded capability offers benefits not only from the perspective of airport resource management but also in various other aspects. Therefore, there is a pressing need for research that focuses on machine learning and neural network models capable of forecasting the distant future using authentic long-term differential data. Hence, in this study, our objective is to specifically address and forecast flight delays of more than 2 h.

Methodology

Classification models

We used the following machine learning models and LSTM neural network to predict flight takeoff delays. The LSTM model boasts the advantage of effectively managing time-series data, but it comes with the drawback of requiring considerably more complex and powerful hardware. From this standpoint, machine learning (ML) models allow predictions at the individual time-unit level and are notably more computationally efficient when compared to the LSTM model.

- Decision Tree (DT): DT is a type of supervised learning model that classifies or regresses data by applying a set of classification rules. The resulting model has a tree-like structure, hence the name 'Decision Tree.' Pruning techniques can be applied

to enhance the model's generalization performance and prevent overfitting, ensuring that it performs effectively on unknown data. Grid search can be used to find the optimal parameter values for the DT model, optimizing its performance [15]. It does not necessitate data preprocessing, such as normalization or handling missing values and outliers. It also has the capability to simultaneously handle both numerical and categorical variables. However, it has the limitation of considering only one variable at a time, which can make it challenging to capture interactions between variables. Moreover, the shape of the resulting decision tree can exhibit significant variations with minor differences in the data [16, 17].

- Random Forest (RF): RF is an ensemble algorithm that trains multiple DT models and combines their results to make predictions. The method entails the random selection of a subset of features from the complete feature set to build one decision tree, followed by the selection of another random feature subset to create additional decision trees. Multiple decision trees are generated using this process. The final prediction is made by choosing the most frequently occurring prediction from these multiple decision trees [18]. This approach is versatile as it can be applied to both classification and regression problems. It is particularly effective in handling large-scale data and mitigates the issue of overfitting by reducing model noise, ultimately improving model accuracy [19, 20].
- Support Vector Machine (SVM): SVM is a powerful supervised learning model that can be used for various tasks such as classification, regression, and anomaly detection. It aims to find a decision boundary that maximizes the separation between two classes while satisfying certain conditions. SVM can handle both linear and non-linear classification problems by using different kernel functions [21]. It determines the side of the decision boundary to which a data point belongs, allowing it to effectively classify data. Although it may be slower and less interpretable due to the requirement for multiple combination tests, it offers the advantage of being applicable to both categorical and numerical prediction problems, with minimal vulnerability to outlier data. Additionally, it is less susceptible to overfitting and more user-friendly compared to neural networks [22, 23].
- K-Nearest Neighbors (KNN): KNN is a classification algorithm that operates based on the principle of similarity. It assigns a class label to a given data point by considering the labels of its "k" nearest neighbors in the feature space. The distance between data points is typically calculated using the Euclidean distance measurement method [24]. It offers several advantages, such as high accuracy and the ability to exclude outlier data from consideration by using only the top k closest data points. Furthermore, it does not rely on assumptions about the data since it is based on existing data. However, it has the disadvantage of increased processing time as the dataset size grows, as it needs to compare with all existing data points, and it may require significant memory usage for large datasets [22, 25].
- Logistic Regression (LR): LR is one of the simplest classification models. It predicts the probability of data belonging to a certain category as a value between 0 and 1 and classifies it into the category with a higher probability [26]. It has the advantage of being less complex and faster due to linear combinations, making it easy to interpret the results. However, it may suffer a reduction in learning ability when dealing with

non-linear relationships and can be sensitive to outliers and anomalies, which are its disadvantages [27, 28].

- Extreme Gradient Boosting (XGB): XGB is an algorithm implemented using the boosting technique. It supports both regression and classification problems and exhibits suitable performance and resource efficiency. It is characterized by strong durability with its built-in overfitting regularization function [29, 30].
- Long Short-Term Memory (LSTM): LSTM networks are a type of RNN that can learn the order dependence in sequence prediction problems. RNNs are modified by adding a memory cell that can store information for an extended period. LSTM was proposed as a solution to address the issue of vanishing gradients in RNN when processing long sequential data [31]. However, it has the drawback of being computationally intensive and having a complex model structure due to the incorporation of forget gates, input gates, and output gates [32–34].

Evaluation methods

To evaluate the performance of each classifier, we calculated the confusion matrix and measured the accuracy, precision, recall, and F-score. Table 4 is the confusion matrix, a 2×2 matrix representation of classification results. The number of correctly classified instances is the sum of the diagonals of the matrix, while all other instances are incorrectly classified. Each item in the confusion matrix includes the following four indicators.

The first indicator is True Positive (TP), which signifies that the predicted value is positive when the actual value is positive. The second indicator is True Negative (TN), indicating that the predicted value is negative when the actual value is negative. The third indicator is False Positive (FP), denoting that the predicted value is positive when the actual value is negative. Lastly, the fourth indicator is False Negative (FN), showing that the predicted value is negative when the actual value is positive [35].

Accuracy serves as "*a metric for assessing the overall performance of each model by computing the ratio of correctly classified samples to the total number of samples*" [36]. However, in situations with a significant imbalance between positive and negative samples, accuracy may not provide a suitable evaluation measure. Precision presents "*the proportion of true positive cases among all predicted positive cases*" [37], while recall computes "*the ratio of correctly predicted positive samples to the total number of true positive samples*" [38]. F1-score represents "*a balanced measure that combines both precision and recall*" [39].

Table 4 Confusion matrix

	Classified as delayed	Classified as not delayed
Actual delayed	True positive (TP)	False negative (FN)
Actual not delayed	False positive (FP)	True negative (TN)

Implementation and result

Data description and analysis

We collected three datasets including flight and weather information of Incheon International Airport in South Korea (ICN) [40], John F. Kennedy International Airport (JFK) [41], and Chicago Midway International Airport (MDW) [42] in the United States.

The flight information [43, 44] is organized by all flight-related features, including scheduled departure time, actual departure time, and delay type. The weather information is the officially introduced regional weather feature. The flight information scheduled from 2010 to 2021 was examined, spanning a total of 11 years. The weather information corresponding to the same period was also collected. For the experiment, weather and flight information were merged with a time difference for data preprocessing to predict flights based on weather conditions. The merged datasets include the attributes listed in Tables 5 and 6. Among these attributes, the airline, flight number, and destination were not used in the actual model training. Additionally, since the features wind direction (e.g., NW, WNW) and condition (e.g., Cloudy, Windy) are categorical data, they were transformed into one-hot encoding before being included in the training dataset.

Data processing

ICN dataset

In situations where the scheduled departure time differs by more than 1 h, we classify the data as delayed. The ICN dataset comprises 1,562,029 instances of normal flights and

Table 5 Incheon International Airport's attributes list

Attribute name	Description	Mean (Std)	Min	Max
Time (year)	2010–2021 (e.g. 2020)	–	–	–
Airline	Unique carrier [e.g. KE (Korean Air)]	–	–	–
Flight number	Flight number (e.g. KE831)	–	–	–
Destination	Destination (e.g. Asiana Airlines)	–	–	–
Planned departure time	Planned departure time (e.g. 10:30)	–	–	–
Actual departure time	Actual departure time (e.g. 11:30)	–	–	–
Result status	Takeoff intime or delay status (e.g. cancellation)	–	–	–
Delay type	Delay type (e.g. weather-snow)	–	–	–
Wind direction (deg)	Wind direction (deg) (e.g. 10)	204.3 (109.7)	0	360
Wind velocity (KT)	Wind velocity (KT) (e.g. 5)	7.4 (4.4)	0	49
Meteorological range	Visible distance (e.g. 10,000)	8311 (2699.4)	0	11000
Cloud cover	Cloud cover (e.g. 3)	2.9 (1.8)	1	9
Cloud form	Cloud form (e.g. 5)	–	–	–
The height of the cloud ceiling (FT)	The height of the cloud ceiling (FT) (e.g. 15,000)	7390.2 (6950.7)	0	24000
Temperature (celcius)	Temperature (celcius) (e.g. − 7)	12.5 (10.49)	− 17.2	36.6
Dew point temperature (celcius)	Dew point temperature (celcius) (e.g. − 18.8)	5.6 (11.4)	− 28.4	26.5
Sea-level pressure (hPA)	The pressure of the atmosphere at the sea level (e.g. 1023.8)	1016.6 (8.4)	981.3	1040.4
Station pressure (hPa)	Station pressure (hPa) (e.g. 1022.8)	1015.8 (8.4)	980.6	1039.5
Rainfall (mm)	Rainfall (mm) (e.g. 0.3)	0.3 (1.4)	0	66

Table 6 John F. Kennedy International Airport, and Chicago Midway International Airport's attributes list

Attribute mame	Description	Mean (Std)	Min	Max
Time (year)	2010–2021 (e.g. 2020)	–	–	–
Airline	Unique carrier [e.g. AA (American Airlines)]	–	–	–
Flight number	Flight number (e.g. AA2000)	–	–	–
Destination	Destination (e.g. JFK)	–	–	–
Planned departure time	Planned departure time (e.g. 1622)	–	–	–
Actual departure time	Actual departure time (e.g. 1634)	–	–	–
Result status	Takeoff intime or delay status (e.g. 1)	–	–	–
Delay type	Delay type (e.g. WeatherDelay)	–	–	–
Wind direction	Wind direction (e.g. NW, WNW)	–	–	–
Wind speed	Wind speed (e.g. 3)	10.5 (5.3)	0	51
Wind gust	Wind gust (e.g. 24)	5.3 (10.9)	0	75
Temperature (celcius)	Temperature (celcius) (e.g. 34)	51.5 (20.5)	− 21	103
Dew point temperature (celcius)	Dew point temperature (celcius) (e.g. 31)	39.9 (19.5)	− 32	79
Humidity	Humidity (e.g. 92)	67.7 (17.2)	0	100
Pressure (hPa)	Pressure (hPa) (e.g. 29.96)	29.3 (0.3)	0	30.2
Precipitation (mm)	Precipitation (mm) (e.g. 0.1)	0.006 (0.046)	0	2
Condition	Condition (e.g. Cloudy, Windy)	–	–	–

7850 instances of delayed flights caused by weather conditions. To achieve a balanced distribution between normal and delayed cases, we randomly sampled an equal number of normal and delayed flight instances. To address the absence of certain features in the cases, we utilized a data interpolation method that was previously validated in a research study [45]. Due to the hourly-based nature of the ICN weather information, there were instances of missing features. To fill these gaps, we employed a linear interpolation technique to estimate the values for the unmeasured time periods. The interpolated data comprises 953 data points, which accounts for 0.9% of the total 105,192 data points. Furthermore, we included flight takeoff results with time differences as additional features. To fulfill the objectives of the present study, we implemented a time difference criterion and utilized combined flight and weather cases. The time differences were categorized into intervals of 2, 4, 8, 16, 24, and 48 h.

JFK dataset and MDW dataset

Similar to the ICN dataset, we created delayed flight instances for the JFK and MDW datasets based on the time difference between the scheduled and actual departure times. The JFK dataset consisted of 763,930 normal cases and 9417 delayed cases attributed to weather conditions, while the MDW dataset comprised 398,945 normal cases and 5562 delayed flight instances. Similar to the approach followed for the ICN dataset, we conducted down-sampling procedures to achieve a 1:1 ratio of normal and delayed cases.

In both the JFK and MDW datasets, the weather information consists of several categorical features, such as wind direction and condition details. To incorporate these features into our data-driven approaches for machine learning and neural network frameworks, we employed a one-hot encoding technique. This encoding method allows us to represent the categorical variables as binary vectors, facilitating their utilization

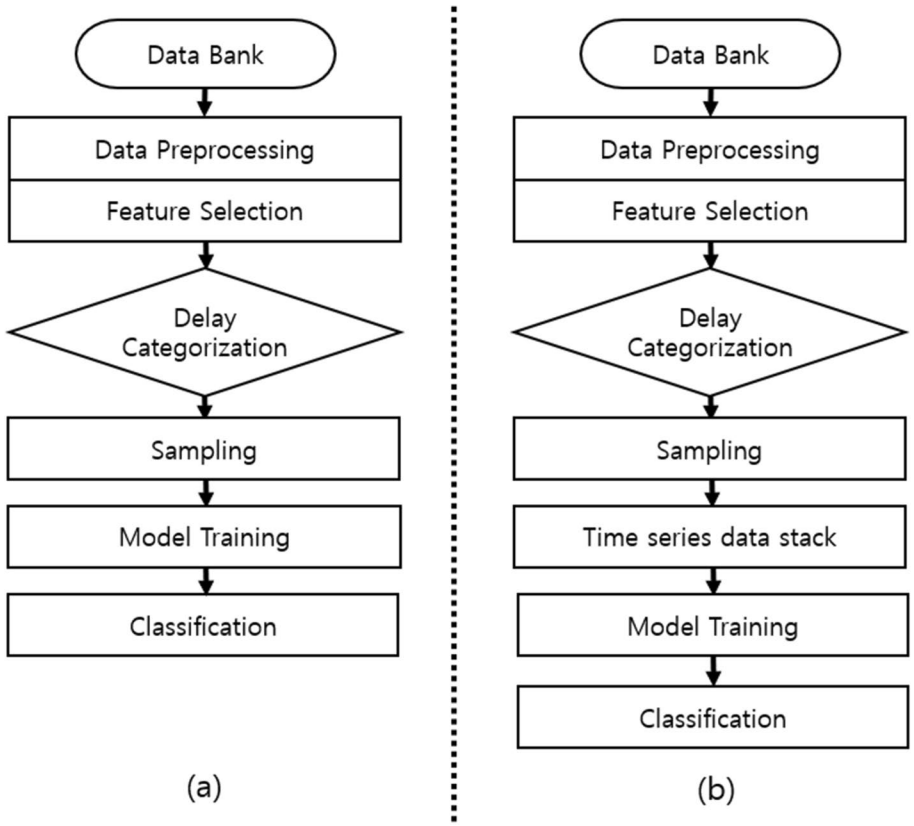

Fig. 1 Flow charts for **a** machine learning, and **b** LSTM models

Table 7 Summary of the employed datasets in training, validation, and test sessions

Airport	Train	Validation	Test	Total
ICN	10551 (67.2%)	2009 (12.8%)	3140 (20.0%)	15700 (100%)
JFK	12756 (68.7%)	2041 (11.0%)	3767 (20.3%)	18564 (100%)
MDW	7476 (68.6%)	1196 (11.0%)	2225 (20.4%)	10897 (100%)

in the models. Additionally, we included flight takeoff results with time differences as one of the features in the dataset. Subsequently, both the JFK and MDW datasets with weather information were merged.

Experiment

Figure 1 shows the flow chart of our overall approach. For machine learning models, we input the data sampled following the process as mentioned above, while we stack the sampled data to create time-series data and input them to the LSTM model.

To begin, we partitioned the dataset into subdata and testing subsets in an 80:20 ratio. Subsequently, we further divided the subdata into training and validation subsets in an 80:20 ratio, resulting in a distribution of the training, validation, and test datasets with a ratio of 67:13:20. Table 7 presents the number of datasets used for training, validation, and testing.

Table 8 Tested parameters in DT

Parameters	Value
Max depth	2, 4, 6, 8, 10, 12, 14, 16, 18, 20
Min impurity decrease	0.0001, 0.0005, 0.001, 0.005, 0.01
Min samples split	2, 3, 4, 5
Min samples leaf	1, 2, 3, 4, 5

Table 9 Tested parameters in LSTM model

Parameters	Value
Layer	1, 2, 3
Learning rate	0.0001, 0.0003, 0.0005, 0.001, 0.005
Epoch	300, 400, 500, 600, 700
Time series	2 h, 3 h, 4 h, 5 h, 6 h

All experiments were conducted on a single GeForce RTX 3080 Ti 10GB GPU and implemented using Python 3.6 as the programming language. We performed a grid search to determine the optimal hyperparameters, including learning rates, number of epochs, number of layers, and number of stacked time-series data. We selected the most optimal parameters for the best performance. Tables 8 and 9 show the list of hyperparameters for DT and LSTM used in the grid search. In the case of the LSTM model, the training parameters varied for each airport dataset. The ICN dataset had 2,385 parameters, while the JFK and MDW datasets had 2,833 parameters.

Results

Flight delay prediction

Tables 10, 11 and 12 show the prediction results of flight departure delays based on weather data using various models. The results were obtained corresponding to a total of six different time differences (2, 4, 8, 16, 24, and 48 h).

Table 10 summarizes the results of the ICN dataset. The RF model reported the highest accuracy score of 0.749 with a time difference of 2 h. Except for the DT model that showed the best recall performance of 0.700, the RF model displayed superior performance in other metrics.

For the JFK airport dataset with a time difference of 2 h, the LSTM model achieved the highest accuracy score of 0.852 (Table 11). In terms of recall for predicting flight delays, the DT model outperformed all other models (0.826), whereas in terms of precision of prediction of on-time flights, the RF model outperformed all other models (0.835). Nonetheless, the LSTM model demonstrated superior performance in other evaluation metrics.

The result corresponding to the MDW airport dataset for a time difference of 2 h is presented in Table 12. The LSTM model achieved the highest accuracy score of 0.785. Although the DT model exhibited the best performance in terms of recall (0.759), the LSTM model outperformed the other models in all other evaluation metrics.

Table 10 Results of ICN airport

Algorithm		Time difference: 2 h						Time difference: 4 h					
		Accuracy	Precision	Recall	F1-score	Train (s)	Test (us)	Accuracy	Precision	Recall	F1-score	Train (s)	Test (us)
DT	Normal	0.688	0.704	0.676	0.690	0.112	0.318	0.681	0.693	0.681	0.687	0.099	0.318
	Delayed		0.671	**0.700**	0.685				0.669	0.681	0.675		
RF	Normal	**0.749**	**0.729**	**0.814**	**0.769**	2.254	16.242	0.735	0.717	0.802	0.757	2.231	16.242
	Delayed		**0.776**	0.680	**0.725**				0.760	0.665	0.710		
SVM	Normal	0.651	0.631	0.774	0.695	3.625	458.280	0.646	0.629	0.756	0.687	3.452	474.522
	Delayed		0.686	0.522	0.593				0.672	0.529	0.592		
KNN	Normal	0.641	0.655	0.637	0.646	0.003	60.510	0.652	0.662	0.661	0.661	0.004	31.847
	Delayed		0.628	0.646	0.637				0.642	0.643	0.643		
LR	Normal	0.595	0.600	0.635	0.617	0.085	0.318	0.583	0.591	0.613	0.602	0.094	0.637
	Delayed		0.589	0.552	0.570				0.575	0.553	0.563		
XGB	Normal	0.721	0.715	0.759	0.736	0.150	1.274	0.707	0.700	0.753	0.725	0.125	1.274
	Delayed		0.728	0.680	0.703				0.716	0.659	0.686		
LSTM	Normal	0.644	0.620	0.776	0.689	490.4	3.503	0.609	0.602	0.679	0.638	490.8	0.318
	Delayed		0.687	0.509	0.584				0.618	0.537	0.575		
Algorithm		**Time difference: 8 h**						**Time difference: 16 h**					
		Accuracy	**Precision**	**Recall**	**F1-score**	**Train (s)**	**Test (us)**	**Accuracy**	**Precision**	**Recall**	**F1-score**	**Train (s)**	**Test (us)**
DT	Normal	0.678	0.691	0.675	0.683	0.095	0.318	0.687	0.702	0.678	0.690	0.091	0.318
	Delayed		0.665	0.681	0.673				0.672	0.696	0.684		
RF	Normal	0.744	0.726	0.806	0.764	2.187	16.242	0.745	0.719	0.826	0.769	2.136	16.242
	Delayed		0.768	0.679	0.721				0.782	0.659	0.715		
SVM	Normal	0.641	0.625	0.750	0.682	3.591	530.255	0.641	0.626	0.749	0.682	3.595	545.541
	Delayed		0.666	0.525	0.587				0.666	0.528	0.589		
KNN	Normal	0.662	0.673	0.666	0.669	0.003	30.892	0.649	0.663	0.643	0.653	0.004	32.803
	Delayed		0.651	0.658	0.655				0.635	0.656	0.645		
LR	Normal	0.598	0.612	0.591	0.601	0.095	0.318	0.525	0.535	0.574	0.554	0.069	0.637
	Delayed		0.583	0.605	0.594				0.512	0.472	0.491		

Table 10 (continued)

Algorithm		Time difference: 8 h						Time difference: 16 h					
		Accuracy	Precision	Recall	F1-score	Train (s)	Test (us)	Accuracy	Precision	Recall	F1-score	Train (s)	Test (us)
XGB	Normal	0.727	0.714	0.784	0.747	0.123	0.955	0.714	0.706	0.759	0.732	0.123	1.274
	Delayed		0.745	0.668	0.704				0.724	0.665	0.693		
LSTM	Normal	0.587	0.581	0.669	0.622	488.4	0.318	0.540	0.531	0.797	0.637	490.1	3.503
	Delayed		0.595	0.502	0.545				0.566	0.274	0.369		
Algorithm		**Time difference: 24 h**						**Time difference: 48 h**					
		Accuracy	**Precision**	**Recall**	**F1-score**	**Train (s)**	**Test (us)**	**Accuracy**	**Precision**	**Recall**	**F1-score**	**Train (s)**	**Test (us)**
DT	Normal	0.676	0.704	0.672	0.688	0.093	0.637	0.7680	0.692	0.679	0.685	0.102	0.637
	Delayed		0.670	0.702	0.685				0.668	0.681	0.674		
RF	Normal	0.743	0.724	0.808	0.764	2.230	16.561	**0.748**	0.721	0.830	0.772	2.465	19.427
	Delayed		0.769	0.674	0.718				0.786	0.661	0.718		
SVM	Normal	0.647	0.625	0.784	0.695	3.922	572.930	0.631	0.619	0.732	0.671	3.846	592.357
	Delayed		0.688	0.502	0.581				0.650	0.525	0.580		
KNN	Normal	0.651	0.652	0.651	0.005	31.529	0.648	0.660	0.650	0.655	0.004	35.032	0.641
	Delayed		0.632	0.630	0.631				0.636	0.646	0.641		
LR	Normal	0.547	0.564	0.528	0.545	0.089	0.637	0.554	0.565	0.574	0.569	0.098	0.637
	Delayed		0.533	0.568	0.550				0.542	0.532	0.537		
XGB	Normal	0.705	0.702	0.739	0.720	0.150	1.274	0.703	0.693	0.758	0.724	0.138	1.274
	Delayed		0.708	0.669	0.688				0.716	0.646	0.679		
LSTM	Normal	0.580	0.586	0.591	0.588	493.3	3.185	0.551	0.548	0.666	0.601	494.4	3.503
	Delayed		0.574	0.569	0.572				0.556	0.433	0.487		

Bold valuesindicate the greatest results

Table 11 Results of JFK airport

Algorithm		Time difference: 2 h						Time difference: 4 h					
		Accuracy	Precision	Recall	F1-score	Train (s)	Test (us)	Accuracy	Precision	Recall	F1-score	Train (s)	Test (us)
DT	Normal	0.787	0.819	0.751	0.783	0.055	0.637	0.790	0.827	0.745	0.784	0.049	0.637
	Delayed		0.759	**0.826**	0.791				0.758	0.837	0.795		
RF	Normal	0.843	**0.835**	0.864	0.849	0.993	21.019	0.850	0.838	0.877	0.857	1.058	20.064
	Delayed		0.852	0.821	0.836				0.864	0.822	0.842		
SVM	Normal	0.650	0.643	0.709	0.675	5.253	618.153	0.638	0.646	0.650	0.648	4.914	632.166
	Delayed		0.658	0.588	0.621				0.630	0.626	0.628		
KNN	Normal	0.712	0.749	0.659	0.701	0.008	70.382	0.722	0.761	0.667	0.711	0.005	40.045
	Delayed		0.682	0.768	0.722				0.691	0.780	0.732		
LR	Normal	0.581	0.597	0.560	0.578	0.107	0.637	0.573	0.594	0.527	0.558	0.118	0.637
	Delayed		0.566	0.603	0.584				0.556	0.622	0.587		
XGB	Normal	0.779	0.783	0.785	0.784	0.164	2.548	0.769	0.772	0.778	0.775	0.127	2.548
	Delayed		0.774	0.772	0.773				0.765	0.760	0.762		
LSTM	Normal	**0.852**	0.831	**0.882**	**0.856**	560.0	4.140	0.829	0.826	0.833	0.829	564.4	4.140
	Delayed		**0.876**	0.822	**0.848**				0.833	0.826	0.830		
Algorithm		**Time difference: 8 h**						**Time difference: 16 h**					
		Accuracy	**Precision**	**Recall**	**F1-score**	**Train (s)**	**Test (us)**	**Accuracy**	**Precision**	**Recall**	**F1-score**	**Train (s)**	**Test (us)**
DT	Normal	0.796	0.826	0.764	0.793	0.049	0.637	0.800	0.827	0.770	0.798	0.051	0.637
	Delayed		0.770	0.831	0.799				0.775	0.831	0.802		
RF	Normal	0.843	0.835	0.865	0.850	1.018	20.382	0.840	0.832	0.863	0.847	1.091	20.064
	Delayed		0.853	0.820	0.836				0.850	0.817	0.833		
SVM	Normal	0.643	0.656	0.635	0.646	5.503	766.242	0.642	0.661	0.618	0.639	6.148	802.548
	Delayed		0.630	0.651	0.640				0.625	0.667	0.645		
KNN	Normal	0.724	0.763	0.669	0.713	0.011	48.726	0.725	0.758	0.681	0.717	0.008	48.726
	Delayed		0.692	0.782	0.735				0.697	0.771	0.733		
LR	Normal	0.594	0.617	0.545	0.579	0.112	0.637	0.582	0.605	0.529	0.564	0.124	0.637
	Delayed		0.575	0.645	0.608				0.563	0.638	0.598		

Table 11 (continued)

Algorithm		Time difference: 8 h						Time difference: 16 h					
		Accuracy	Precision	Recall	F1-score	Train (s)	Test (us)	Accuracy	Precision	Recall	F1-score	Train (s)	Test (us)
XGB	Normal	0.776	0.785	0.776	0.780	0.132	2.548	0.778	0.783	0.782	0.783	0.121	2.548
	Delayed		0.767	0.777	0.772				0.772	0.773	0.772		
LSTM	Normal	0.814	0.829	0.790	0.809	565.3	4.140	0.799	0.773	0.843	0.807	565.1	4.140
	Delayed		0.802	0.838	0.820				0.829	0.756	0.791		
Algorithm		**Time difference: 24 h**						**Time difference 48 h**					
		Accuracy	**Precision**	**Recall**	**F1-score**	**Train (s)**	**Test (us)**	**Accuracy**	**Precision**	**Recall**	**F1-score**	**Train (s)**	**Test (us)**
DT	Normal	0.779	0.816	0.733	0.772	0.056	0.637	0.784	0.821	0.740	0.779	0.051	0.955
	Delayed		0.747	0.826	0.785				0.753	0.831	0.790		
RF	Normal	0.837	0.822	0.869	0.845	1.081	21.975	**0.846**	0.830	0.880	0.854	1.067	20.064
	Delayed		0.854	0.803	0.827				0.865	0.811	0.837		
SVM	Normal	0.618	0.637	0.589	0.612	6.033	824.841	0.625	0.641	0.608	0.624	6.403	890.127
	Delayed		0.601	0.649	0.624				0.610	0.644	0.626		
KNN	Normal	0.723	0.755	0.678	0.714	0.008	51.911	0.721	0.752	0.681	0.714	0.011	43.949
	Delayed		0.695	0.770	0.730				0.695	0.764	0.728		
LR	Normal	0.562	0.593	0.463	0.520	0.125	0.637	0.565	0.588	0.504	0.543	0.112	0.637
	Delayed		0.542	0.666	0.598				0.547	0.629	0.585		
XGB	Normal	0.778	0.787	0.777	0.782	0.121	2.229	0.773	0.777	0.782	0.779	0.123	2.548
	Delayed		0.769	0.779	0.774				0.770	0.764	0.767		
LSTM	Normal	0.778	0.780	0.771	0.776	568.2	4.140	0.736	0.724	0.761	0.742	569.4	4.140
	Delayed		0.776	0.785	0.780				0.748	0.710	0.729		

Bold valuesindicate the greatest results

Table 12 Results of MDW airport

Algorithm		Time difference: 2 h						Time difference: 4 h					
		Accuracy	Precision	Recall	F1-score	Train (s)	Test (us)	Accuracy	Precision	Recall	F1-score	Train (s)	Test (us)
DT	Normal	0.731	0.741	0.702	0.721	0.073	0.955	0.722	0.750	0.659	0.702	0.045	0.955
	Delayed		0.721	**0.759**	0.740				0.701	0.784	0.740		
RF	Normal	0.762	0.748	0.784	0.766	0.997	17.834	0.766	0.750	0.792	0.771	1.011	17.197
	Delayed		0.777	0.741	0.759				0.784	0.741	0.762		
SVM	Normal	0.587	0.588	0.558	0.573	3.716	542.994	0.600	0.596	0.599	0.598	3.715	564.968
	Delayed		0.586	0.615	0.600				0.604	0.601	0.602		
KNN	Normal	0.642	0.645	0.620	0.632	0.007	73.248	0.646	0.649	0.623	0.636	0.009	29.936
	Delayed		0.640	0.664	0.652				0.643	0.668	0.656		
LR	Normal	0.571	0.570	0.548	0.558	0.079	0.637	0.581	0.579	0.567	0.573	0.076	0.955
	Delayed	0.594	0.582				0.583	0.595	0.589				0.578
XGB	Normal	0.716	0.718	0.703	0.710	0.178	2.866	0.715	0.717	0.702	0.709	0.142	3.185
	Delayed		0.714	0.728	0.721				0.713	0.728	0.720		
LSTM	Normal	**0.785**	**0.755**	**0.849**	**0.799**	404.2	2.548	0.756	0.744	0.786	0.765	411.7	2.548
	Delayed		**0.824**	0.719	**0.768**				0.769	0.725	0.746		
Algorithm		**Time difference: 8 h**						**Time difference: 16 h**					
		Accuracy	**Precision**	**Recall**	**F1-score**	**Train (s)**	**Test (us)**	**Accuracy**	**Precision**	**Recall**	**F1-score**	**Train (s)**	**Test (us)**
DT	Normal	0.716	0.731	0.677	0.703	0.042	0.955	0.726	0.752	0.667	0.707	0.046	1.274
	Delayed		0.704	0.755	0.729				0.705	0.783	0.742		
RF	Normal	0.767	0.75	0.775	0.767	1.070	17.834	0.774	0.752	0.812	0.781	1.331	17.516
	Delayed		0.774	0.758	0.766				0.800	0.737	0.767		
SVM	Normal	0.613	0.619	0.574	0.595	4.172	612.102	0.615	0.619	0.584	0.601	3.862	637.898
	Delayed		0.609	0.652	0.630				0.612	0.646	0.629		
KNN	Normal	0.643	0.657	0.585	0.619	0.006	27.389	0.667	0.677	0.629	0.652	0.006	29.618
	Delayed		0.632	0.701	0.664				0.659	0.705	0.681		

Table 12 (continued)

Algorithm		Time difference: 8 h						Time difference: 16 h					
		Accuracy	Precision	Recall	F1-score	Train (s)	Test (us)	Accuracy	Precision	Recall	F1-score	Train (s)	Test (us)
LR	Normal	0.578	0.577	0.554	0.565	0.071	1.911	0.605	0.605	0.587	0.596	0.080	0.637
	Delayed		0.572	0.601	0.589				0.606	0.624	0.615		
XGB	Normal	0.714	0.723	0.687	0.704	0.140	3.185	0.726	0.731	0.706	0.719	0.150	3.185
	Delayed		0.707	0.741	0.723				0.721	0.745	0.733		
LSTM	Normal	0.741	0.736	0.759	0.747	409.3	2.548	0.718	0.701	0.772	0.735	410.8	2.548
	Delayed		0.746	0.723	0.735				0.741	0.664	0.700		
Algorithm		**Time difference: 24 h**						**Time difference: 48 h**					
		Accuracy	**Precision**	**Recall**	**F1-score**	**Train (s)**	**Test (us)**	**Accuracy**	**Precision**	**Recall**	**F1-score**	**Train (s)**	**Test (us)**
DT	Normal	0.703	0.747	0.683	0.713	0.045	0.955	0.727	0.744	0.684	0.713	0.049	0.955
	Delayed		0.712	0.773	0.741				0.712	0.769	0.740		
RF	Normal	0.773	0.762	0.787	0.774	1.081	18.153	**0.772**	0.759	0.790	0.774	1.128	16.879
	Delayed		0.784	0.758	0.771				0.785	0.754	0.769		
SVM	Normal	0.600	0.611	0.528	0.566	3.974	595.223	0.614	0.619	0.577	0.597	4.377	630.573
	Delayed		0.591	0.670	0.628				0.610	0.652	0.630		
KNN	Normal	0.663	0.667	0.641	0.654	0.006	27.389	0.664	0.679	0.609	0.642	0.008	26.433
	Delayed		0.660	0.685	0.672				0.651	0.717	0.683		
LR	Normal	0.597	0.601	0.558	0.579	0.084	0.955	0.596	0.598	0.563	0.580	0.096	0.637
	Delayed		0.594	0.635	0.614				0.594	0.628	0.611		
XGB	Normal	0.731	0.737	0.710	0.723	0.134	3.503	0.725	0.732	0.703	0.717	0.152	3.503
	Delayed		0.725	0.751	0.738				0.719	0.747	0.733		
LSTM	Normal	0.714	0.697	0.768	0.731	415.2	2.548	0.712	0.719	0.702	0.710	416.1	2.548
	Delayed		0.736	0.659	0.696				0.705	0.722	0.713		

Bold valuesindicate the greatest results

Flight delay prediction (1 to 24 h, hourly)

Tables 13, 14 and 15 provide an hourly breakdown of model accuracy from 1 h to 24 h, utilizing the same three datasets for ICN, JFK, and MDW airports, along with average training and testing times. The hyperparameters that yielded the best performance in the prior experiments were applied. Across all three airport datasets, the highest accuracy was observed at a 1-h time difference, with a declining trend in performance as the time difference increased. The magnitude of performance decline from 1 h to 24 h for each model is detailed in Table 16. Notably, the Random Forest model exhibited the least performance degradation, with a decrease of only −3.6%, while the SVM model showed the most significant performance decline, with an average decrease of −16.1%. Machine learning models completed their training in just a few seconds, while LSTM required several 100 s, indicating it was approximately 100 times more time-consuming. In terms of testing time, it ranged from as low as 1 ms to a maximum of around 1.3 ms.

Ablation study

We conducted training on the ICN dataset with identical parameters and training strategies, except for the exclusion of linear interpolation, while examining a time difference of 2 h. The results, as depicted in Table 17, reveal a slight reduction in overall performance, ranging from 1 to 2%, when interpolation was omitted. It is noteworthy that the interpolated data constitutes only 0.9% (953 out of 105,192) of the entire dataset, which lends credibility to the decision to incorporate linear interpolation in our research.

Feature importance

To determine the features with a substantial impact on our models, we conducted feature importance analysis. We chose the Random Forest and LSTM models, which demonstrated the best performance. For the Random Forest model, we made use of the built-in feature importance function, whereas for the LSTM model, we employed external algorithms using loss data. Consequently, in the case of Random Forest, higher values correspond to greater feature importance, whereas for LSTM, lower values signify reduced importance. Considering the results of the ICN airport dataset, Random Forest attributed the highest importance to temperature, dew point, and weather phenomena in that order, while LSTM assigned the highest importance to temperature, wind speed, weather phenomena, and local pressure. Notably, temperature was identified as the most crucial feature in both models (Table 18).

For the JFK airport dataset, Random Forest identified pressure, temperature, and dew point as the most important features, while LSTM emphasized pressure, precipitation, and wind speed as the top influential factors. Notably, pressure was recognized as the most crucial feature in both models for this dataset (Table 19).

In the case of the MDW airport dataset, Random Forest indicated that pressure, humidity, and temperature were the top features in terms of importance, while LSTM emphasized pressure, precipitation, and wind speed as the most influential factors. Notably, pressure was consistently identified as the most important feature in both models for this dataset (Table 20).

Comparison with prior approaches

We conducted a performance comparison between our models and a prior research model [8]. Using the same JFK airport dataset, we compared our research's Random Forest and LSTM models with the prior research model's LSTM model. Our Random Forest model achieved an accuracy of 84.3% with a 2-h time difference and 84.6% with a 48-h time difference. In contrast, the LSTM model in our research achieved an accuracy of 85.2% with a 2-h time difference and 73.6% with a 48-h time difference. It's worth noting that the previous model exhibited a performance of 86.51% at a short time interval of 15 min.

Discussion and concluding remarks

For predicting flight takeoff delays using weather information for the airports of ICN, JFK, and MDW, machine learning and LSTM models were employed. Based on the prediction results for the three regions, the RF model demonstrated the highest performance for the ICN airport, while the LSTM model exhibited the highest performance for JFK and MDW airports, with a minimum time difference of 2 h. The accuracy scores were 0.749 for ICN, 0.852 for JFK, and 0.785 for MDW airports. Moreover, the RF model also displayed the best performance with high accuracy for all three airports, with a maximum time difference of 48 h; the accuracy scores were 0.748 for ICN, 0.846 for JFK, and 0.772 for MDW airports. Moreover, when assessing test times, all of the models require less than 2 ms, which makes them suitable for real-time predictions. These findings confirm the feasibility of predicting flight takeoff delays using weather data collected 2 h prior to the scheduled departure time.

Our analysis incorporated datasets spanning from 2011 to 2021, encompassing a long time period. This extensive dataset allowed us to leverage both actual flight operation data and weather information for our analysis. By utilizing these comprehensive datasets, our proposed models exhibited outstanding performance in predicting delayed flights across three different datasets. The utilization of a long-term dataset facilitated robust predictions and enhanced the reliability of our models. Furthermore, the approaches we developed can be applied to various other transportation-related domains, including ocean vessel delays, vehicle operation restrictions, and outdoor construction work stoppages. In these application areas, early-stage warnings play a crucial role in mitigating potential risks to human safety and property damage. By leveraging our proposed models, it becomes feasible to anticipate and prepare for potential disruptions, enabling proactive measures to be taken in advance. This can significantly contribute to minimizing the adverse impacts associated with delays and restrictions in these transportation-related sectors. The presented implications notwithstanding, it is important to acknowledge the presence of notable limitations. One such limitation is the significant influence of national and regional factors on weather conditions, rendering it challenging to generalize the results to other locations. The generalization of findings beyond the specific context may not be straightforward owing to these variations. Furthermore, the performance of the ICN airport dataset was relatively lower compared with the JFK and MDW airport datasets. This discrepancy in performance could be attributed to several factors, including the presence of missing features in the dataset. The absence

Table 13 Results of ICN airport (accuracy from 1 to 24 h)

Algorithm	1 h	2 h	3 h	4 h	5 h	6 h	7 h	8 h	9 h	10 h	11 h	12 h	13 h
DT	0.688	0.688	0.687	0.681	0.686	0.678	0.674	0.678	0.677	0.675	0.680	0.675	0.661
RF	0.750	0.749	0.748	0.735	0.746	0.738	0.741	0.744	0.751	0.743	0.743	0.751	0.739
SVM	0.697	0.651	0.686	0.646	0.653	0.654	0.652	0.641	0.660	0.645	0.641	0.638	0.640
KNN	0.659	0.641	0.654	0.652	0.652	0.646	0.646	0.662	0.651	0.641	0.644	0.635	0.638
LR	0.669	0.595	0.659	0.583	0.589	0.592	0.582	0.598	0.571	0.573	0.547	0.537	0.548
XGB	0.734	0.721	0.733	0.707	0.715	0.726	0.718	0.727	0.717	0.714	0.712	0.722	0.712
LSTM	0.622	0.644	0.609	0.609	0.624	0.602	0.579	0.587	0.573	0.580	0.571	0.562	0.539
Algorithm	**14 h**	**15 h**	**16 h**	**17 h**	**18 h**	**19 h**	**20 h**	**21 h**	**22 h**	**23 h**	**24 h**	**Avg train (s)**	**Avg test (us)**
DT	0.686	0.677	0.687	0.675	0.680	0.689	0.684	0.682	0.685	0.680	0.687	0.011	0.318
RF	0.737	0.743	0.745	0.747	0.739	0.739	0.756	0.745	0.738	0.746	0.743	3.613	20.701
SVM	0.643	0.628	0.641	0.636	0.651	0.651	0.628	0.640	0.641	0.640	0.647	6.295	970.382
KNN	0.643	0.630	0.649	0.653	0.629	0.629	0.641	0.655	0.640	0.657	0.641	0.004	47.134
LR	0.521	0.568	0.525	0.554	0.527	0.527	0.528	0.546	0.554	0.544	0.547	0.074	0.637
XGB	0.715	0.710	0.714	0.714	0.715	0.715	0.716	0.713	0.706	0.716	0.705	0.211	1.592
LSTM	0.558	0.563	0.540	0.564	0.561	0.561	0.571	0.571	0.548	0.545	0.580	468.7	184.713

Table 14 Results of JFK airport (accuracy from 1 to 24 h)

Algorithm	1 h	2h	3 h	4 h	5 h	6 h	7 h	8 h	9 h	10 h	11 h	12 h	13 h
DT	0.831	0.787	0.828	0.790	0.826	0.806	0.804	0.796	0.789	0.791	0.791	0.790	0.799
RF	0.882	0.843	0.877	0.850	0.869	0.853	0.854	0.843	0.846	0.848	0.845	0.842	0.846
SVM	0.801	0.650	0.782	0.638	0.728	0.701	0.698	0.643	0.662	0.678	0.671	0.665	0.638
KNN	0.803	0.712	0.792	0.722	0.772	0.756	0.738	0.724	0.731	0.739	0.736	0.724	0.730
LR	0.706	0.581	0.665	0.573	0.642	0.617	0.606	0.594	0.582	0.584	0.605	0.588	0.560
XGB	0.859	0.779	0.851	0.769	0.822	0.801	0.801	0.776	0.788	0.783	0.786	0.782	0.779
LSTM	0.848	0.852	0.842	0.829	0.828	0.826	0.817	0.814	0.809	0.812	0.804	0.803	0.800
Algorithm	**14 h**	**15 h**	**16 h**	**17 h**	**18 h**	**19 h**	**20 h**	**21 h**	**22 h**	**23 h**	**24 h**	**Avg train (s)**	**Avg test (us)**
DT	0.799	0.790	0.800	0.786	0.797	0.784	0.781	0.796	0.797	0.784	0.779	0.081	0.796
RF	0.844	0.844	0.840	0.836	0.844	0.832	0.841	0.839	0.839	0.846	0.837	1.809	21.503
SVM	0.646	0.659	0.642	0.620	0.650	0.616	0.653	0.651	0.628	0.610	0.618	11.877	1289.089
KNN	0.727	0.724	0.725	0.728	0.724	0.708	0.722	0.727	0.722	0.707	0.723	0.009	57.606
LR	0.577	0.565	0.582	0.575	0.577	0.559	0.603	0.608	0.569	0.557	0.562	0.128	0.531
XGB	0.774	0.779	0.778	0.772	0.783	0.771	0.786	0.779	0.769	0.767	0.778	0.194	2.655
LSTM	0.795	0.809	0.799	0.804	0.803	0.796	0.794	0.786	0.785	0.785	0.778	554.2	4.513

Table 15 Results of MDW airport (accuracy from 1 to 24 h)

Algorithm	1 h	2 h	3 h	4 h	5 h	6 h	7 h	8 h	9 h	10 h	11 h	12 h	13 h
DT	0.755	0.731	0.743	0.722	0.763	0.740	0.738	0.716	0.745	0.726	0.731	0.743	0.720
RF	0.811	0.762	0.787	0.766	0.791	0.783	0.803	0.767	0.789	0.767	0.779	0.783	0.783
SVM	0.735	0.587	0.716	0.600	0.690	0.651	0.664	0.613	0.627	0.616	0.642	0.630	0.606
KNN	0.728	0.642	0.727	0.646	0.710	0.675	0.698	0.643	0.661	0.692	0.688	0.674	0.667
LR	0.698	0.571	0.655	0.581	0.641	0.612	0.607	0.578	0.566	0.595	0.590	0.589	0.575
XGB	0.797	0.716	0.769	0.715	0.776	0.741	0.750	0.714	0.731	0.715	0.755	0.743	0.745
LSTM	0.785	0.785	0.769	0.756	0.773	0.746	0.744	0.741	0.750	0.749	0.737	0.727	0.734
Algorithm	**14 h**	**15 h**	**16 h**	**17 h**	**18 h**	**19 h**	**20 h**	**21 h**	**22 h**	**23 h**	**24 h**	**Avg train (s)**	**Avg test (us)**
DT	0.717	0.731	0.726	0.728	0.726	0.693	0.719	0.730	0.731	0.709	0.728	0.052	0.899
RF	0.777	0.776	0.774	0.773	0.765	0.756	0.766	0.790	0.782	0.756	0.773	1.243	23.820
SVM	0.584	0.615	0.615	0.596	0.608	0.572	0.600	0.622	0.563	0.580	0.600	4.500	884.045
KNN	0.661	0.663	0.667	0.659	0.665	0.653	0.646	0.658	0.670	0.652	0.663	0.008	48.090
LR	0.546	0.584	0.605	0.574	0.580	0.587	0.581	0.604	0.577	0.559	0.597	0.089	0.899
XGB	0.732	0.723	0.726	0.714	0.717	0.707	0.725	0.741	0.728	0.700	0.731	0.177	4.045
LSTM	0.716	0.728	0.718	0.724	0.713	0.729	0.723	0.715	0.709	0.727	0.714	391.6	3.596

Table 16 Comparison of accuracy levels between 1 and 24 h

Algorithm	ICN (%)	JFK (%)	MDW (%)	Average (%)
DT	− 1.7	− 6.3	− 6.9	− 5.0
RF	− 0.9	− 5.1	− 4.7	− 3.6
SVM	− 7.2	− 22.8	− 18.4	− 16.1
KNN	− 2.7	− 10.0	− 8.9	− 7.2
LR	− 18.2	− 20.4	− 14.5	− 17.7
XGB	− 4.0	− 9.4	− 8.3	− 7.2
LSTM	− 6.8	− 8.3	− 9.0	− 10.1

Table 17 Ablation study on linear interpolation in the ICN dataset with a time difference of 2 h

Algorithm		With linear interpolation				Without linear interpolation			
		Accuracy	Precision	Recall	F1-score	Accuracy	Precision	Recall	F1-score
DT	Normal	0.688	0.704	0.676	0.690	0.680	0.695	0.672	0.683
	Delayed		0.671	0.700	0.685		0.665	0.688	0.677
RF	Normal	0.749	0.729	0.814	0.769	0.740	0.724	0.800	0.760
	Delayed		0.776	0.680	0.725		0.762	0.678	0.718
SVM	Normal	0.651	0.631	0.774	0.695	0.600	0.597	0.684	0.638
	Delayed		0.686	0.522	0.593		0.605	0.511	0.554
KNN	Normal	0.641	0.655	0.637	0.646	0.635	0.643	0.653	0.648
	Delayed		0.628	0.646	0.637		0.627	0.616	0.622
LR	Normal	0.595	0.600	0.635	0.617	0.545	0.553	0.590	0.571
	Delayed		0.589	0.552	0.570		0.534	0.496	0.514
XGB	Normal	0.721	0.715	0.759	0.736	0.688	0.681	0.737	0.708
	Delayed		0.728	0.680	0.703		0.696	0.635	0.664

Table 18 Feature importance of ICN airport

Feature importance	Wind speed	Visibility	Weather phenomena	Temperature	Dew point	Sea-level pressure	Local pressure
RF	0.062	0.057	0.073	**0.078**	0.076	0.071	0.069
LSTM	20.718	20.724	20.721	**20.715**	21.006	20.932	20.721

Bold valuesindicate the greatest results

Table 19 Feature importance of JFK airport

Feature importance	Temperature	Dew point	Humidity	Wind speed	Wind gust	Pressure	Precipitation
RF	0.183	0.167	0.155	0.109	0.037	**0.184**	0.005
LSTM	68.161	22.531	43.579	10.509	35.987	**8.548**	9.584

Bold valuesindicate the greatest results

of these features may have impacted the overall performance of the models. Future research endeavors should focus on addressing these limitations by exploring more comprehensive datasets and improving data collection methods to minimize missing features. This would enhance the generalizability and accuracy of the models in predicting flight delays.

Table 20 Feature importance of MDW airport

Feature importance	Temperature	Dew point	Humidity	Wind speed	Wind gust	Pressure	Precipitation
RF	0.165	0.158	0.170	0.109	0.029	**0.176**	0.004
LSTM	36.448	30.777	26.572	13.893	78.303	**11.887**	11.957

Bold valuesindicate the greatest results

In future research, our aim is to develop a more robust model that incorporates geographic information, enabling its application to other airports beyond the specific datasets analyzed in this study.

Author contributions
Kim contributed to the design, implementation, and analysis of the research with the examination of the manuscript. Kim and Park wrote and revised the manuscript. Park approved the final version of the manuscript.

Funding
This work was supported by the Ministry of Education of the Republic of Korea and the National Research Foundation of Korea (NRF-2023S1A5A8075518). This research was supported by the MSIT (Ministry of Science and ICT), Korea, under the ICAN (ICT Challenge and Advanced Network of HRD) support program (IITP-2023-RS-2023-00259497) supervised by the IITP (Institute for Information & Communications Technology Planning & Evaluation).

Availability of data and materials
The datasets used and/or analysed during the current study are available from the corresponding author on reasonable request.

Declarations

Ethics approval and consent to participate
Not applicable.

Consent for publication
Not applicable.

Competing interests
The authors declare that they have no competing interests.

Received: 9 June 2023 Accepted: 14 December 2023
Published online: 09 January 2024

References

1. Economics-IATA: air passenger market analysis 2014. 2014.
2. Economics-IATA: air passenger market analysis 2023. 2023. https://www.iata.org/en/iata-repository/publications/economic-reports/air-passenger-market-analysis2/.
3. Efthymiou M, Njoya ET, Lo PL, Papatheodorou A, Randall D. The impact of delays on customers' satisfaction: an empirical analysis of the British airways on-time performance at Heathrow airport. J Aerosp Technol Manag. 2018;11:e0219.
4. Tileagă C, Oprisan O. Flights delay compensation 261/2004: a challenge for airline companies? In: Organizations and Performance in a complex world: 26th international economic conference of Sibiu (IECS) 26. Springer; 2021. p. 335–44.
5. U.S.D. of transportation, airline on-time performance data. 2023. https://www.transtats.bts.gov/tables.asp?QO_VQ=EFD &QO_anzr=Nv4yv0r.
6. Sim G-M, Kim Y-S, Jung M-P, Kim J-W, Park M-S, Hong S-H, Kang K-K. Changes in the frequency of abnormal weather events in South Korea in recent years. J Korean Soc Clim Change. 2018;9(4):461–70.
7. Lee J-W, Yoo H-I, Kim G-H. Analysis of South Korea's heavy rain characteristics from 2006 to 2015 using AWS data. In: Proceedings of the Korean meteorological society conference. 2016. p. 521–2.
8. Kim YJ, Choi S, Briceno S, Mavris D. A deep learning approach to flight delay prediction. In: 2016 IEEE/AIAA 35th digital avionics systems conference (DASC). IEEE; 2016. p. 1–6.
9. Khaksar H, Sheikholeslami A. Airline delay prediction by machine learning algorithms. Scientia Iranica. 2019;26(5):2689–702.
10. Qu J, Wu S, Zhang J. Flight delay propagation prediction based on deep learning. Mathematics. 2023;11(3):494.
11. Al-Tabbakh SM, El-Zahed H. Machine learning techniques for analysis of Egyptian flight delay. J Sci Res Sci. 2018;35(part 1):390–9.

12. Ye B, Liu B, Tian Y, Wan L. A methodology for predicting aggregate flight departure delays in airports based on supervised learning. Sustainability. 2020;12(7):2749.
13. Atlioğlu MC, Bolat M, Şahin M, Tunali V, Kilinç D. Supervised learning approaches to flight delay prediction. Sakarya Univ J Sci. 2020;24(6):1223–31.
14. Yazdi MF, Kamel SR, Chabok SJM, Kheirabadi M. Flight delay prediction based on deep learning and Levenberg–Marquart algorithm. J Big Data. 2020;7:1–28.
15. Lee J, Cha J, Park E. Data-driven approaches into political orientation and news outlet discrimination: the case of news articles in south korea. Telemat Inform. 2023;85: 102066.
16. Gao Z, Gatpandan MP, Gatpandan PH. Classification decision tree algorithm in predicting students' course preference. In: 2021 2nd international symposium on computer engineering and intelligent communications (ISCEIC). IEEE; 2021. p. 93–7.
17. Sharma A, Sharma M, Dwivedi R. Improved decision tree classification (IDT) algorithm for social media data. In: 2021 10th international conference on system modeling & advancement in research trends (SMART). IEEE; 2021. p. 155–7.
18. Kim E, Ji H, Kim J, Park E. Classifying apartment defect repair tasks in South Korea: a machine learning approach. J Asian Archit Build Eng. 2022;21(6):2503–10.
19. Soumya A, Kumar GH. Classification of ancient epigraphs into different periods using random forests. In: 2014 fifth international conference on signal and image processing. IEEE; 2014. p. 171–8.
20. Ardiansyah D, Mantoro T, Syafei WA. Potential classification prediction of solar and wind energy in Indonesia using machine learning with random forest algorithm. In: 2022 5th international conference of computer and informatics engineering (IC2IE). IEEE; 2022. p. 297–302.
21. Lee J, Park E. D-HRSP: dataset of helpful reviews for service providers. Telemat Inform. 2023;82:102001.
22. Fadhil IM, Sibaroni Y. Topic classification in Indonesian-language tweets using fast-text feature expansion with support vector machine (SVM). In: 2022 international conference on data science and its applications (ICoDSA). IEEE; 2022. p. 214–9.
23. Charan PVS, Ramkumar G. Black fungus classification using Adaboost with SVM-based classifier and compare accuracy with support vector machine. In: 2022 5th international conference on contemporary computing and informatics (IC3I). IEEE; 2022. p. 1895–901.
24. Hwang S, Ahn H, Park E. iMovieRec: a hybrid movie recommendation method based on a user-image-item model. Int J Mach Learn Cybern. 2023;14:3205–16.
25. Auleria M, Arrahmah AI, Saputra DE. A review on KN nearest neighbour based classification for object recognition. In: 2021 international conference on data science and its applications (ICoDSA). 2021; IEEE. p. 274–80.
26. Kim S, An C, Cha J, Kim D, Park E. D-visa: a dataset for detecting visual sentiment from art images. In: Proceedings of the IEEE/CVF international conference on computer vision. 2023. p. 3051–9.
27. Akoulih M, Tigani S, Saadane R, Tazi A. Electrocoagulation based chromium removal efficiency classification using logistic regression. Appl Sci. 2020;10(15):5179.
28. Guan X, Zhang J, Chen S. Logistic regression based on statistical learning model with linearized kernel for classification. Comput Inform. 2021;40(2):298–317.
29. Paleczek A, Grochala D, Rydosz A. Artificial breath classification using XGBoost algorithm for diabetes detection. Sensors. 2021;21(12):4187.
30. Liang H, Li J, Wu H, Li L, Zhou X, Jiang X. Mammographic classification of breast cancer microcalcifications through extreme gradient boosting. Electronics. 2022;11(15):2435.
31. Lee S, Jeong D, Park E. MultiEmo: multi-task framework for emoji prediction. Knowl-Based Syst. 2022;242: 108437.
32. Hur Y. Malaysian name-based ethnicity classification using LSTM. KSII Trans Internet Inf Syst. 2022;16(12):3855–67.
33. Zerrouki N, Houacine A, Harrou F, Bouarroudj R, Cherifi MY, Sun Y. Exploiting deep learning-based LSTM classification for improving hand gesture recognition to enhance visitors' museum experiences. In: 2022 international conference on innovation and intelligence for informatics, computing, and technologies (3ICT). IEEE; 2022. p. 451–6.
34. Madanan M, Venugopal A, Velayudhan NC. A hybrid anomaly based intrusion detection methodology using IWD for LSTM classification. In: 2020 IEEE international conference on advanced networks and telecommunications systems (ANTS). IEEE; 2020. p. 1–5.
35. Lee S, Kim J, Kim D, Kim KJ, Park E. Computational approaches to developing the implicit media bias dataset: assessing political orientations of nonpolitical news articles. Appl Math Comput. 2023;458:128219.
36. Lee S, Kim J, Park E. Can book covers help predict bestsellers using machine learning approaches? Telemat Inform. 2023;78: 101948.
37. Park E. CRNet: a multimodal deep convolutional neural network for customer revisit prediction. J Big Data. 2023;10(1):1–10.
38. Oh S, Ji H, Kim J, Park E, del Pobil AP. Deep learning model based on expectation–confirmation theory to predict customer satisfaction in hospitality service. Inform Technol Tour. 2022;24(1):109–26.
39. Yu H, Park E. A harmless webtoon for all: an automatic age-restriction prediction system for webtoon contents. Telemat Inform. 2023;76: 101906.
40. Incheon airport weather. https://data.kma.go.kr/data/air/selectAmosRltmList.do?pgmNo=575 &tabNo=1.
41. New York City weather. https://www.wunderground.com/history/daily/us/ny/new-york-city/KLGA.
42. Chicago City weather. https://www.wunderground.com/history/daily/us/il/chicago/KMDW.
43. Incheon air port flight. https://www.airport.kr/co/ko/cpr/statisticOfDelay.do.
44. United States Department of Transport. https://www.transtats.bts.gov/tables.asp?QO_VQ=EFD &QO_anzr=Nv4yv0r.
45. Panda B, Adhikari RK. A method for classification of missing values using data mining techniques. In: 2020 international conference on computer science, engineering and applications (ICCSEA). IEEE; 2020. p. 1–5.

Publisher's Note