

RouteRisk: Using Historical Flight Data to Identify and Predict Delay-Prone Routes

Team 054: ZhiJun Li, Scott London, Rolfson Aruljothi, Pramodh Aryasomayajula, Alexander Tang, Aaqib

Bickiya

1 Introduction

Delays and cancellations have been one of the largest cost drivers for the air transportation industry. A 2022 study by AirHelp estimated the economic cost of delays and cancellations at 67.5 billion dollars globally. For travelers, these issues can lead to loss of opportunity at the destination, loss in brand trust, or preference for other forms of travel. To grow the industry and maintain customer trust, it is imperative to offer travelers tools that better communicate risks associated with their choice of flight and carrier. RouteRisk will help travelers choose less delay-prone flights, thereby improving the travel experience while reducing cost to carriers from delay compensation and missed connections.

2 Problem Definition

We will use Bureau of Transportation Statistics (BTS) flight data to identify which routes and carriers experience the fewest delays. We will accomplish this through interactive visuals that feature a map showing airports and flight routes in addition to the ability to predict whether a given flight will be delayed.

In more technical terms, we believe there is a lack of useful insights and intuitive visuals for travelers looking to identify the flight with the smallest chance of delay. Our work plans to address this gap through visualizations that begin with a graph structure that represents airports as nodes and routes as edges. Information related to the percentage of delays and volume of flights will be conveyed through the color and scale of graph elements. We will use this visual structure as a common input mechanism for additional analysis tools that will help a user determine:

- (1) Delay Prediction: The chance of a flight being delayed and the likely cause.
- (2) Delay Propagation: Whether a route delay is composed of upstream or downstream effects.

- (3) Carrier Reliability: The likelihood of individual carriers to be delayed for a given route.

The first two analysis tools will be powered by processing BTS data into new tables. The third analysis tool will be powered by a binary classification model.

Delay Prediction

Let \mathcal{X} represent the multidimensional feature space of flight characteristics, and let Y represent the binary target variable. We define a delayed flight as one where the actual arrival time exceeds the scheduled arrival time by 15 minutes or more. Thus, $Y \in \{0, 1\}$, where:

- $Y = 1$ denotes a **delayed** flight (≥ 15 minutes).
- $Y = 0$ denotes an **on-time** flight (< 15 minutes).

For any flight instance i , the input vector $x^{(i)} \in \mathcal{X}$ is defined as a concatenation of feature subsets:

$$x^{(i)} = [x_{time}, x_{route}, x_{env}, x_{status}]^T$$

Where:

- **Temporal** (x_{time}): Features derived from schedule data, including month, day of week, departure period, and holiday proximity.
- **Route** (x_{route}): Categorical identifiers for carrier, origin, and destination, along with great-circle distance.
- **Environmental** (x_{env}): Real-time adverse weather flags and historical route-specific weather delay rates.
- **Status** (x_{status}): Observed gate departure delay (set to 0 for pre-booking forecasts).

Objective: Our objective is to learn a hypothesis function $f_\theta : \mathcal{X} \rightarrow [0, 1]$ that estimates the conditional probability of a delay:

$$\hat{p}(x^{(i)}) = P(Y = 1 \mid X = x^{(i)})$$

The system applies a decision threshold $\tau = 0.5$ to output the final prediction $\hat{Y}^{(i)}$:

$$\hat{Y}^{(i)} = \mathbb{I}(\hat{p}(x^{(i)}) \geq 0.5)$$

where \mathbb{I} is the indicator function.

Delay Propagation

Let $G = (V, E)$ be a directed graph where V is the set of airports and E is the set of routes. For a given aircraft with tail number t on date d , its rotation sequence is an ordered list of flights $\langle f_1^{(t,d)}, f_2^{(t,d)}, \dots, f_k^{(t,d)} \rangle$ sorted by scheduled departure time. For consecutive legs f_{j-1} and f_j , we define the inherited delay as the arrival delay of the preceding leg:

$$\delta_{\text{inherited}}(f_j) = \text{arr_delay}(f_{j-1})$$

For a route $r = (o, d) \in E$ within time window $w \in \{\text{morning, afternoon, evening}\}$, let $\mathcal{L}_{r,w}$ be the set of all consecutive-leg pairs where the current leg operates on route r during window w , with $|\mathcal{L}_{r,w}| \geq 30$. The mean inherited delay for route r in window w is:

$$\bar{\delta}_{\text{inh}}(r, w) = \frac{1}{|\mathcal{L}_{r,w}|} \sum_{(f_{j-1}, f_j) \in \mathcal{L}_{r,w}} \delta_{\text{inherited}}(f_j)$$

Each upstream link is classified by risk tier ρ :

$$\rho(r, w) = \begin{cases} \text{low} & \text{if } \bar{\delta}_{\text{inh}}(r, w) < 10 \text{ min} \\ \text{moderate} & \text{if } 10 \leq \bar{\delta}_{\text{inh}}(r, w) \leq 25 \text{ min} \\ \text{high} & \text{if } \bar{\delta}_{\text{inh}}(r, w) > 25 \text{ min} \end{cases}$$

Carrier Reliability

For a route $r = (o, d)$ and carrier c , let $\mathcal{F}_{r,c}$ denote the set of all flights operated by carrier c on route r . The carrier delay rate is:

$$D(r, c) = \frac{|\{f \in \mathcal{F}_{r,c} : \text{arr_del15}(f) = 1\}|}{|\mathcal{F}_{r,c}|}$$

The carrier reliability score combines delay rate with volume to penalize carriers with both high delay rates and high traffic:

$$S(r, c) = D(r, c) \times \sqrt{|\mathcal{F}_{r,c}|}$$

For a given route, the optimal carrier recommendation is:

$$c^*(r) = \arg \min_{c : |\mathcal{F}_{r,c}| \geq n_{\min}} D(r, c)$$

3 Literature Survey

We reviewed relevant work while planning RouteRisk that can broadly be categorized into two categories: Visualizing Flight Data and Forecasting Delays.

Visualizing Flight Data

To portray the relationship between airports and to visualize volume and traffic, we researched prior work to see how others were working with similar datasets. Burch et al. [3] and Chen et al. [5] both have utilized a graph structure to great effect for this use case. Methods such as heat maps and bubble tree maps were evaluated, but did not perform as well in user studies. Zhu et al. [18] build on the idea of using graphs by introducing techniques such as conditional independence tests to model delay propagation within graph networks. All three works were designed for the aviation industry and need changes to transfer these insights to a visual that suits general audiences.

Forecasting Delays

Our dashboard will feature the ability to select nodes and forecast delay based on history and other variables set by the user. We researched relevant work to understand machine learning techniques that can help us create accurate predictions.

Dhanawade et al. [8] highlight critical features to consider when forecasting delay and provide us guidance on how to transform data prior to use. Al-Bassam et al. [1] and Yi et al. [17] highlight the success of data balancing techniques such as random oversampling and SMOTE to provide the delayed flight class with a sufficient amount of data. Loy [14] provides an initial analysis of how popular machine learning techniques perform when predicting flight delays. We initially considered simple regression techniques, but Li et al. [13] mention potential risk with this method due to the role of the larger airport network and how delay is diffused. Tan et al. [16] shows how we can alleviate the risk Li mentioned by using his ANSP score to quantify

the effects of delay within the larger network. Hyndman et al. [10] cautions against improper splitting and testing of time series when using forecasting algorithms.

Güvercin et al. [9] and Jacyna-Golda et al. [11] introduce novel techniques for forecasting delays with the clustered K-means approach and score-card system respectively. These techniques provide guidance on structuring data for use with other machine learning techniques. Bisandu et al. [2] and Taecharungroj et al. [15] also show novel methodologies by incorporating deep learning into the process. Looking at more classic methods, Chakrabarty et al. [4] and Kiliç et al. [12] demonstrate the success of gradient boosting algorithms for this use case. Similarly, Dai [7] and Chen et al. [6] show how Random Forest classification techniques can be successful at predicting if a flight will be delayed or not. Many of these experiments were conducted with limited datasets which leaves room for growth on larger datasets with multiple carriers.

4 Proposed Method

RouteRisk will provide an interactive visualization of historical carrier and route data that provides actionable insights and powers delay predictions. It will consist of a graph visual showing airports and flight routes and three analysis layers: Delay Propagation, Carrier Reliability, and Delay Prediction. Key innovations are:

- (1) Responsive Hybrid Canvas/SVG Rendering
- (2) Delay Propagation Analysis and Visualization
- (3) Multi-Stage Composite ML Architecture for Sparse Event Prediction

Intuition

We examined both commercial offerings and academic work. Commercial offerings are limited to only showing current airport traffic and do not consider historical data. RouteRisk will allow users to visualize and filter through 13.9 million rows of historical data. Commercial tools also do not offer any deeper insight beyond which airports are currently seeing delays. We have prepared analysis views that allow you to predict delay based on flight scenarios, understand which carriers are most prone to delay, and understand how delays

propagate in the broader airport network. Our academic research was focused on innovations in visualizing airport networks and predicting delay. We combined Chet et al.’s [5] research into displaying airport networks and delays in graph form with Burch et al.’s methods of using graphs as control interfaces for other analysis tools. We surveyed a variety of approaches for forecasting delays and compared the performance of different models and techniques. Our final delay classification model utilizes a combination of techniques researched such as SMOTE and Gradient Boosting.

Responsive Hybrid Canvas/SVG Rendering

RouteRisk renders a graph of 355 airports and over 6,000 routes with hover and click interactions. Querying the 13.9M-row flights table on every interaction offered a poor user experience. Other solutions we researched clustered groups of airports to avoid rendering issues, but we prioritized preserving the full dataset which led to a unique two-phase implementation.

An offline computation pipeline creates per-airport and per-route statistics into slim database tables. Per-airport metrics include mean delays, delay cause attribution across four BTS categories (carrier, weather, NAS, late aircraft), and a composite score that accounts for percentage delayed and volume. Routes are classified into four risk tiers: low ($< 15\%$ delayed), medium ($15\text{--}25\%$), high ($25\text{--}40\%$), and severe ($> 40\%$). These tables power all runtime queries, reducing response times to single-digit milliseconds. JSON artifacts are fetched once per session and cached in memory.

For the frontend, we utilize D3.js v7 to draw edges on an HTML5 Canvas for efficient batch drawing of routes, and nodes as SVG circles for native hover/click handling on airports. Geographic positions are projected once via `geoAlbersUsa()` and cached. Edge opacity follows a logarithmic scale from flight count to $[0.08, 0.5]$, and node radius follows a square-root scale from total flights to $[3, 18]$ pixels. The analysis layers are implemented through a plugin system where each layer extends a `BaseLayer` class, and a `LayerManager` enforces mutual exclusion so only one layer is active at a time.

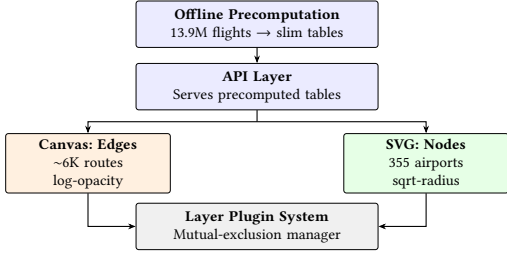


Figure 1: Hybrid Canvas/SVG rendering architecture.

Delay Propagation Analysis and Visualization

A significant proportion of delays are inherited from preceding legs rather than being caused during the flight. The BTS `tail_number` field enables us to account for multi-leg flights. A SQL window function orders flights by `(tail_number, fl_date, crs_dep_time)`, with consecutive legs joined on `leg_num = prev.leg_num + 1`. Each flight is bucketed into either morning (06:00-12:00), afternoon (12:00-18:00), or evening (18:00-6:00) windows to account for plane usage throughout the day. For each combination of route and time window with at least 30 observations, we compute the mean inherited arrival delay from preceding legs and classify the link as low, moderate, or high risk. The downstream analysis reverses direction, aggregating departure delays of subsequent legs. A verdict system compares delay rates across time windows to recommend optimal booking periods with carrier recommendations. The frontend visualizes delay chains by drawing upstream feeders as colored dashed lines with animated directional dots.

Multi-Stage Composite ML Architecture for Sparse Event Prediction

Our delay prediction engine is a binary classification system determining whether a flight will exceed a 15-minute arrival delay. To transition from a static model to a functional pipeline, we implemented a three-stage architecture: robust feature engineering and leakage prevention using 500,000 stratified samples with temporal features (holiday proximity, binned departure periods), weather indicators (per-route historical delay rates), and categorical variables, while excluding all post-flight

columns to ensure real-world validity; imbalance-aware training via an 80/20 stratified split combined with Synthetic Minority Over-sampling Technique (SMOTE) and class-weighted loss functions to address delay sparsity; and model selection and optimization by benchmarking five classifiers with F1 score as the primary metric and ROC-AUC as a tiebreaker.

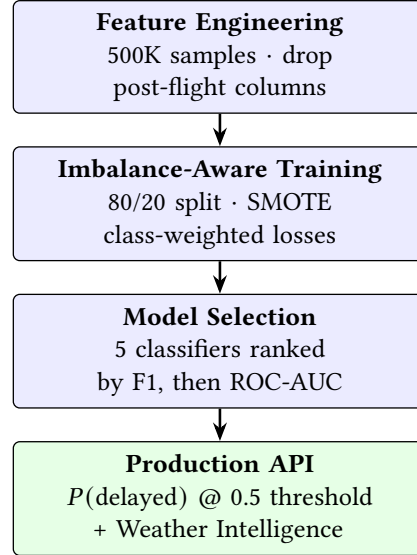


Figure 2: Three-stage ML pipeline for delay prediction.

An optimized Gradient Boosting model (200 trees, $\eta = 0.05$, depth = 4) outperformed the other approaches as shown by Table 1. The production inference API mirrors the training-time feature pipeline in real time, outputting $P(\text{delayed})$ at a 0.5 threshold, and includes a “Weather Intelligence” layer that flags adverse conditions when historical route-month weather delay rates exceed 15%.

Table 1: Model comparison on test set (100,000 samples).

Model	F1	ROC-AUC
Baseline (majority)	0.0000	0.5000
Logistic Regression	0.7775	0.9272
Random Forest	0.8118	0.9300
Gradient Boosting	0.8131	0.9312
XGBoost	0.7741	0.9317

5 Evaluation

Our work consists of three distinct analysis layers that utilize an underlying graph visual as a control mechanism. Our evaluations are designed to quantify if the visuals are intuitive to users, if RouteRisk meaningfully reduces the amount time lost to delays, and if we are able to accurately predict delays.

Testbed Questions

- Do users find RouteRisk intuitive to use and are they able clearly derive and understand desired insights?
- Were RouteRisk users able to find an alternative flight to their destination that experienced less delay than the flight they would taken normally.
- Are flight delay predictions accurate on unseen data?

Experiments

Question 1: Usability Test

To evaluate the usability of RouteRisk, we will recruit 50 test participants with varied travel experience and organize a controlled test session. Each participant will be asked to complete a list of 10 tasks of similar difficulty using various RouteRisk features. An evaluator will be present to record statistics about the test and provide assistance if asked for. After completing all tasks, participants will be asked to complete the System Usability Scale (SUS).

Evaluators will collect the following metrics:

- (1) The mean amount of time spent per task
- (2) The percentage of tasks completed without assistance across all participants
- (3) The SUS score derived from post-session survey.

The mean time per task will mainly be used to identify areas of improvement rather than benchmark success. To consider this experiment successful, we would like to see at least 80 percent of tasks completed and SUS score of 70. Both marks would be above average, anything lower will result in additional improvements being prioritized. Due to time constraints impacting user facing studies, a full evaluation will be conducted in the future. To gather and share initial observations, we conducted

the test described above with 5 individuals with ages spanning from 20s to 50s who travel on flights at least twice a year. The tests returned a mean time spent per task of 54.74 seconds, a task completion rate of 77.2%, and a SUS score of 82. While it is premature to draw conclusions from the limited sample size, we are tracking below our task completion rate goal, and above our SUS score goal. Early feedback from users focused on the unintuitive naming for analysis layers, filters, and certain data points. Multiple testers remarked that they did not know what "Propagation Chain" meant and that the layer should be described differently.

Question 2: Delay Reduction Test

To evaluate whether RouteRisk helps users experience a statically significant reduction in time lost to delays, we will recruit 50 additional participants with similar qualifications to the first test. Each participant will be given a random travel scenario and asked to select an appropriate flight as they would normally (null flight). After picking an initial flight, the tester will be asked to use RouteRisk to choose a flight for the given scenario (alternative flight). A proctor will be present to ensure that the tester is able to use RouteRisk without issue and that usability problems do not leak into the test. Once the second flight has been selected, the tester will be dismissed.

We will compare the delay of each paired null flight and alternative flight once data is available. A paired t-test will determine whether the mean delay reduction is statistically significant at $\alpha = 0.05$. We will report the effect size using Cohen's d to quantify significance. To control for scenario difficulty, we will stratify results by historical rate of delay using RouteRisk's built-in categories (low, medium, high, severe) and verify that improvements are consistent across tiers rather than driven by a subset of easy scenarios.

Due to time constraints impacting user facing studies, a full evaluation will be conducted in the future. To gather and share initial observations, we conducted a simplified version of the test with 10 flight scenarios. Eight scenarios had a negligible difference between departure times while two scenarios experienced an average of 6 minutes faster

departure than scheduled. Though we cannot draw conclusions due to sample size, we considered this a positive sign for the full test.

Question 3: Model Evaluation

To evaluate if RouteRisk’s delay predictions are accurate, we utilized a temporal holdout test to determine how the binary classification model performs on unseen data. This setup replicates RouteRisk’s real-world usage when predicting future flights. The model was trained on all flight data from January 2023 through September 2024 and evaluated on a holdout set spanning October 2024 through December 2024, comprising 1,781,482 flights (1,493,518 on-time, 287,964 delayed).

We define success for this evaluation with three metrics. First, the delayed-class F1 score must exceed 0.70, ensuring the model maintains a meaningful balance between catching real delays and avoiding false alarms. Second, ROC-AUC must remain above 0.90 and show the ability to distinguish classes. Lastly, precision for the delayed class must be above 85% so that user trust is not eroded by incorrect predictions.

Table 2: Delay Prediction Model Performance on Oct–Dec 2024 data

Metric	Value
Accuracy	0.9378
Precision	0.8895
Recall	0.7024
F1 Score	0.7850
ROC-AUC	0.9254

Table 3: Per-class Model Performance on Oct–Dec 2024 data

Class	Precision	Recall	F1-Score
0 (On-time)	0.94	0.98	0.96
1 (Delayed)	0.89	0.70	0.78
Accuracy			0.94
Macro Avg	0.92	0.84	0.87
Weighted Avg	0.94	0.94	0.93

The model achieved an overall accuracy of 93.78% and a ROC-AUC of 0.9254, showing a strong ability to differentiate delayed and on-time flights months into the future. For the delayed class, precision reached 88.95% with an F1 Score of 0.78. For on-time flights, the model achieved an F1 score of 0.96 with 98% recall. These results show that users can be confident when RouteRisk labels a flight as low-risk or high-risk. These results achieved all evaluation criteria we aimed for and confirm that the model generalizes to unseen data and will provide trustworthy predictions to users.

6 Conclusions and Discussion

RouteRisk utilizes a graph representation of flight data and novel algorithms to produce intuitive visuals and insights. Our goal is for the everyday traveler to avoid unexpected delays, which will lead to downstream benefits for carriers and a healthier air transportation industry.

While user studies are still pending, the positive results related to our delay prediction capabilities show that RouteRisk will immediately provide value to travelers looking for a smoother flying experience. Being able to accurately predict delay will help users understand risk associated with their choice of flight and compare against other options. We will use the future results of the user studies to further refine our tools and interface.

While our initial implementation has already returned promising results, we have identified room for further improvement. Price is a critical factor for travelers when choosing flights that we do include in our visualization. A 2022 survey by Airlines for America reported that 46% of travelers believe that price is the most important factor when choosing a flight. Our value proposition for users is predicated on the belief that travelers value a smoother travel experience, but it may not be at the cost of a significantly more expensive flight. A future version could integrate either live or average prices for flights and include a composite score that ranks flights by a combination of price and risk of delay.

All group members contributed equally to this project

References

- [1] Sarah Ahmed A AlBassam and Dhafir N AlShahrani. 2025. Flight delay prediction: Evaluating machine learning algorithms for enhanced accuracy. *PLoS One* 20, 12 (2025), e0335141.
- [2] D. B. Bisandu and I. Moulitsas. 2023. A Deep BiLSTM Machine Learning Method for Flight Delay Prediction Classification. *Journal of Aviation/Aerospace Education & Research* 32, 2 (2023).
- [3] Michael Burch, Anna Kuriger, Franjo Pehar, and Bernard Bekavac. 2025. Flights and Airports-A Visual Analytics Perspective. In *2025 MIPRO 48th ICT and Electronics Convention*. IEEE, 1558–1563.
- [4] Navoneel Chakrabarty. 2019. A data mining approach to flight arrival delay prediction for american airlines. In *2019 9th annual information technology, Electromechanical Engineering and Microelectronics Conference (IEMECON)*. IEEE, 102–107.
- [5] C. Chen, C. Li, J. Chen, and C. Wang. 2022. VFDP: Visual Analysis of Flight Delay and Propagation on a Geographical Map. *IEEE Transactions on Intelligent Transportation Systems* 23, 4 (2022), 3510–3521.
- [6] Jun Chen and Meng Li. 2019. Chained predictions of flight delay using machine learning. In *AIAA SciTech Forum*. https://junchen.sdsu.edu/proceedings/scitech_gnc19_Chen.pdf
- [7] M. Dai. 2024. A hybrid machine learning-based model for predicting flight delay through aviation big data. *Scientific Reports* 14 (2024), 4603.
- [8] Rutuja Dhanawade, Mandar Deo, Nidhi Khanna, and Rugved V Deolekar. 2019. Analyzing factors influencing flight delay prediction. In *2019 6th International Conference on Computing for Sustainable Global Development (INDIACom)*. IEEE, 1003–1007.
- [9] M. Güvercin, N. Ferhatosmanoglu, and B. Gedik. 2021. Forecasting Flight Delays Using Clustered Models Based on Airport Networks. *IEEE Transactions on Intelligent Transportation Systems* 22, 5 (2021), 3179–3190.
- [10] Rob J. Hyndman and George Athanasopoulos. 2021. *Forecasting: Principles and practice* (3rd ed.). OTexts. <https://otexts.com/fpp3/tscv.html>
- [11] Ilona Jacyna-Golda, Krzysztof Cur, Justyna Tomaszewska, Karol Przanowski, Sarka Hoskova-Mayerova, and Szymon Świergolik. 2025. Optimizing Flight Delay Predictions with Scorecard Systems. *Applied Sciences* 15, 11 (2025), 5918.
- [12] Kerim Kiliç and Jose M Sallan. 2023. Study of delay prediction in the US airport network. *Aerospace* 10, 4 (2023), 342.
- [13] Q. Li, L. Wu, X. Guan, and Z. Tian. 2024. Interplay of network topologies in aviation delay propagation: A complex network and machine learning analysis. *Physica A: Statistical Mechanics and its Applications* 638 (2024), 129622.
- [14] Dong Xuan Loy. 2024. Flight data analysis and delay prediction system. *Applied Information Technology And Computer Science* 5, 1 (2024), 475–492.
- [15] P. Taecharungroj et al. 2021. Flight Delay Prediction Using a Hybrid Deep Learning Method. *Engineering Journal* 25, 8 (2021), 99–112.
- [16] Yi Tan, Yajun Lu, and Lu Wang. 2025. Flight delay dynamics: Unraveling the impact of airport-network-spilled propagation on airline on-time performance. *Decision Support Systems* 196 (2025), 114494. <https://doi.org/10.1016/j.dss.2025.114494>
- [17] Jia Yi, Honghai Zhang, Hao Liu, Gang Zhong, and Guiyi Li. 2021. Flight delay classification prediction based on stacking algorithm. *Journal of Advanced Transportation* 2021, 1 (2021), 4292778.
- [18] Dan Zhu, Huawei Wang, and Xianghua Tan. 2024. Mining delay propagation causality within an airport network from historical data. *Aerospace* 11, 7 (2024), 533.