Flight Prices Prediction

Anh Nguyen, Fiona Xu, and Sadichchha Maharjan

Abstract— **Accurate flight ticket price prediction remains a complex challenge due to the highly dynamic and nonlinear nature of airline pricing strategies. While traditional machine learning models such as K-Nearest Neighbors (KNN), Random Forest, and Decision Trees have been widely employed, their ability to capture temporal dependencies is limited. This project investigates the effectiveness of deep learning models, specifically Gated Recurrent Units (GRU) and Long Short-Term Memory (LSTM) networks, in modeling sequential patterns in airfare data. Using a dataset of nearly 6 million U.S. domestic flight records collected between April and November 2022, we compare the performance of these recurrent neural network models against baseline methods such as Linear Regression and XGBoost. Our results are unfortunately unable to indicate whether deep learning models can offer improvements in prediction accuracy. However, they highlight the potential to capture time-based trends and dependencies, which can ultimately contribute to more robust and generalizable airfare forecasting solutions.**

I. Introduction

Flight pricing is influenced by a variety of factors such as booking date, departure time, demand surges, competition, and route characteristics. Due to these complexities, building reliable airfare prediction models remains a persistent challenge in both academia and industry. Traditional machine learning models, including Decision Trees, Random Forests, and K-Nearest Neighbors (KNN), have been widely applied to this task and have demonstrated reasonable performance in modeling non-linear relationships between input features and ticket prices.

However, these conventional models are often limited in their ability to model temporal dependencies—key to understanding how prices evolve over time and in response to market fluctuations. As a result, there is increasing interest in applying deep learning techniques that are specifically designed to capture sequential patterns in time-series data.

In this project, we evaluate whether recurrent neural networks (RNNs), particularly GRU and LSTM architectures, can provide performance gains in airfare prediction by effectively leveraging the temporal structure of flight pricing data. We use a large-scale dataset of nearly 6 million flight records from U.S. airlines, spanning from April to November 2022, to compare the performance of these deep learning models against established baselines: Linear Regression and XGBoost. Our objectives are to improve prediction accuracy, optimize feature selection, and reduce the risk of overfitting, thereby contributing to more dependable airfare forecasting tools.

II. Previous Works

Airfare prediction has been extensively studied using traditional machine learning (ML) models such as K-Nearest Neighbors (KNN), Decision Trees, Random Forests, and XGBoost. These models have demonstrated effectiveness in modeling static relationships between features like departure time, number of stops, and airline type. For instance, Groves and Gini employed ensemble methods to enhance prediction accuracy [1], while Joshi et al. utilized Random Forests to capture complex feature interactions [2].

However, these traditional models often fall short in capturing the temporal dependencies inherent in flight pricing data. To address this, recent research has explored deep learning approaches, particularly Recurrent Neural Networks (RNNs) and their variants. Degife and Lin proposed a GRU-based model that significantly outperformed classic ML models in predicting flight fares, highlighting deep learning algorithms' capability to model sequential data effectively [3].

Further advancements include integrating sentiment analysis with deep learning models. Degife and Lin later introduced a hybrid model combining aspect-based sentiment analysis (ABSA) with GRU networks, leveraging customer reviews alongside transactional data to enhance fare prediction accuracy [4].

These studies underscore the potential of deep learning models, especially GRUs and LSTMs, in capturing the dynamic and temporal aspects of airfare data, offering improvements over traditional ML approaches.

III. Dataset

The dataset used for this project was sourced from Kaggle, named Flight Prices [5]. The CSV file contains flight ticket listings scraped from Expedia between April 16, 2022, and October 5, 2022, covering routes between 15 major U.S. airports, including LAX, JFK, EWR, SFO, BOS, IAD, and others. The scheduled flight dates in the dataset range from April 17 through November 11, 2022. The dataset contains 5,999,739 unique entries, with each row representing a single, purchasable flight ticket . Each entry includes 27 features that capture information related to pricing, scheduling, and flight characteristics.

These features include information such as the date the ticket was searched, the scheduled flight date, origin and destination airports, airline details, and pricing attributes such as the base fare and total fare. The dataset also includes flags for whether a flight ticket is refundable, basic economy, or nonstop, as well as the number of seats remaining. Travel duration, travel distance, and time stamps for departures and arrivals are also included.

To better understand the pricing distribution, a histogram of the totalFare variable was plotted, with values limited to the 95th percentile to reduce the influence of extreme outliers.

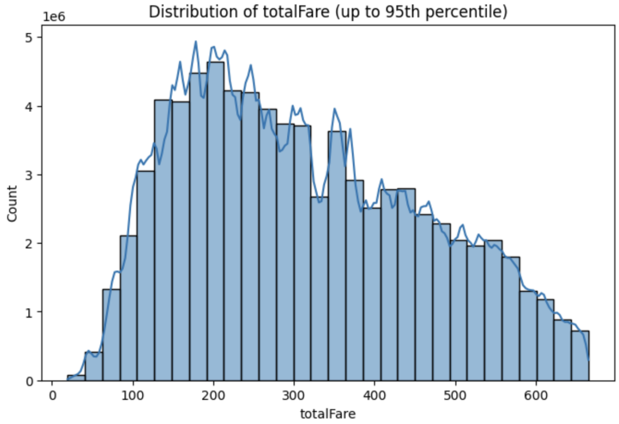


Fig. 1: Distribution of totalFare for flight tickets, capped at the 95th percentile to visualize typical fare ranges.

As illustrated in Fig. 1, the distribution of totalFare is right-skewed, with the majority of ticket prices concentrated between $100 and $300. This indicates that most fares fall within a moderate price range, while a long tail of higher-priced tickets is also present. Understanding this distribution is critical for informing preprocessing decisions and shaping an effective modeling strategy for flight price prediction.

To further support the analysis, the trend of average ticket prices over time was visualized. The time series plot in Fig. 2 reveals clear seasonal patterns, with prices rising from May to mid-July before gradually declining through November. These fluctuations likely reflect variations in demand driven by holidays, travel seasons, and other temporal factors. Recognizing such trends emphasizes the importance of incorporating time-based features into model to capture price dynamics better.

This extensive dataset, with its rich variety of features and large volume of entries, provides a strong foundation for training deep learning models aimed at predicting flight prices. By leveraging detailed travel-related factors and temporal trends within the data, the models can better understand the complex dynamics influencing fare fluctuations. Data preprocessing techniques, which are discussed in the next section, were necessary to prepare the features for modeling, ensure data consistency, and handle missing values.

IV. Methods

A. Data Preprocessing

Data preprocessing involved removing irrelevant columns, handling missing values, and engineering additional features relevant to flight pricing. Due to the dataset's size, preprocessing was parallelized using PySpark. A total of 11 columns, including ID fields and redundant information, were removed.

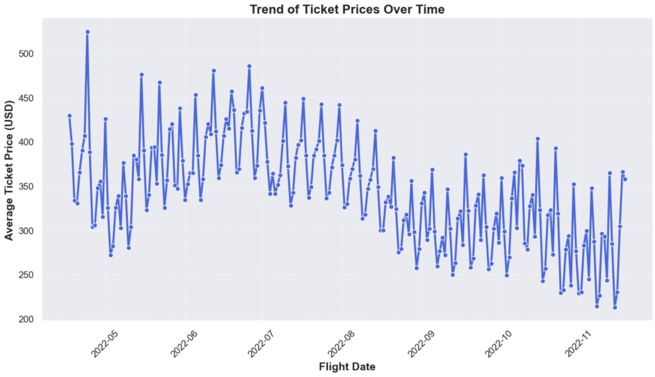


Fig. 2: Average ticket prices over time, highlighting seasonal fluctuations and pricing trends from May to November 2022.

New boolean and numerical features—such as *isCoach*, *days_until_flight*, *isHoliday*, and *isWeekend*—were introduced based on domain heuristics. Missing values in *days_until_flight* were imputed using the median.

The dataset was chronologically split by *departure_date* into training (70%), validation (15%), and test (15%) sets. Appropriate transformations were then applied: binary encoding for boolean features, one-hot encoding for categorical variables, and Min-Max scaling for numerical attributes. Twelve features were transformed in total. Finally, the processed data was partitioned into smaller chunks and stored for efficient model training.

B. Baseline Models

Two traditional machine learning models were implemented as baselines for comparison: Linear Regression and XGBoost.

Linear Regression: A standard multivariate linear regression model was trained using the engineered features. This approach provides both predictions and statistical insight into feature significance through coefficient estimates and p-values. Although limited in capturing non-linear relationships, the model offers interpretability and serves as a useful benchmark.

XGBoost: XGBoost is a ensemble machine learning model that builds decision trees where each tree corrects the mistake from the previous one. XGBoost is recognized for its computational efficiency, scalability, and strong performance on structured data, particularly in domains involving high-dimensional feature spaces and large-scale datasets—such as those encountered in flight prediction tasks. [6]

In this study, an XGBoost model was employed as a baseline. The model was configured with 1000 estimators, and early stopping was applied with a patience of 50 rounds based on stagnation in validation loss. This strategy was implemented to prevent overfitting and to enhance the model's generalization capability.

C. Deep Learning Models

To capture temporal dependencies in flight pricing, we implemented two recurrent neural network (RNN) models: GRU and LSTM.

GRU (Gated Recurrent Unit): The GRU (Gated Recurrent Unit) model is a type of recurrent neural network that captures temporal dependencies in sequential data. GRU models are particularly useful for time-series forecasting as they memorize and retain long-term dependencies in sequences similar to LSTMs (Long Short-Term Memory networks). However, in comparison to LSTMs, GRUs are less computationally expensive and faster to train. This made GRUs well-suited for our project, which involved over 5 million flight entries and required learning complex temporal patterns.

Building an effective GRU model to predict time series data is not a simple task. A model with a simple architecture might not be robust enough to capture the nonlinear trend in fluctuating features, while complex models with more layers pose an increased risk of overfitting and longer training times. [7] Therefore, it is important to carefully tune the model's depth and parameters in order to achieve reliable and efficient predictions. To explore this balance, we built and tested both simpler GRU models with fewer layers and more complex versions with additional layers.

To guide the training process and reduce model complexity, we selected nine key input features identified as important by the Linear Regression model, based on their statistical significance and contribution to prediction accuracy. This informed feature selection allowed the GRU to focus on the most relevant attributes without being overwhelmed by noise from less impactful variables. To further stabilize training, the target variable (flight price) was scaled from 0 to 1. Predictions were rescaled to their original range for final evaluation using real-world metrics. Model performance including MSE, MAE, and R² for both variants are discussed in next section.

LSTM (Long Short-Term Memory):

Long Short-Term Memory (LSTM) networks are a specialized class of recurrent neural networks (RNNs) designed to model sequential data by capturing long-range temporal dependencies. Due to their gated architecture, LSTMs effectively address the vanishing gradient problem inherent in traditional RNNs, making them well-suited for time series forecasting tasks.

In this study, three LSTM architectures were developed and evaluated, each differing in complexity and parameterization to investigate the impact of model capacity on forecasting performance. Early stopping was incorporated into all training procedures to prevent overfitting, based on validation loss stagnation.

The first LSTM model comprised two stacked LSTM layers, each with 50 hidden units, and was trained using a subset of 3 temporal features selected from the full set of 55 input features. This configuration served as a baseline LSTM implementation with moderate complexity.

The second model was a low-parameter architecture designed to assess performance under reduced model capacity. It consisted of two LSTM layers with only 4 units each and was trained using the same 9 time-related features as the GRU baseline model.

The third model represented the most complex LSTM configuration, incorporating four LSTM layers with 40 units each. To introduce regularization and improve generalization, dropout with a rate of 0.5 was applied to two of the layers. This model also utilized the same 9 time series features as the GRU model.

Among the three, the third LSTM architecture achieved the highest overall performance, demonstrating the benefit of increased model complexity and regularization in capturing temporal dependencies present in the dataset.

V. Evaluation

Model performance was assessed using three standard regression metrics: Mean Squared Error (MSE), Mean Absolute Error (MAE), and the Coefficient of Determination (R^2). These metrics provide complementary perspectives on model accuracy and error characteristics.

MSE Measures the average squared difference between actual and predicted values. It penalizes larger errors more severely and is sensitive to outliers. MAE calculates the average absolute difference between actual and predicted values, offering a more interpretable and robust error measure. R^2 represents the proportion of variance in the target variable that is predictable from the features. Higher values indicate a better model fit.

All metrics were computed on a chronologically separated test set to evaluate generalization performance.

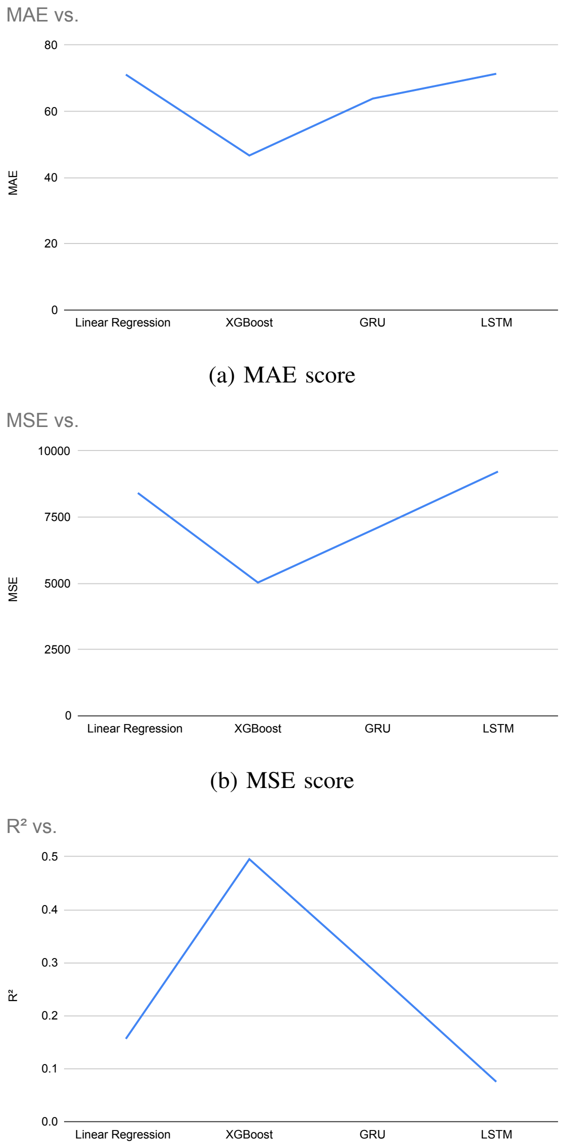
VI. Results

The performance of four predictive models—Linear Regression, XGBoost, GRU, and LSTM—was evaluated using three key metrics: Mean Squared Error (MSE), Mean Absolute Error (MAE), and the Coefficient of Determination (R^2).

Among all models, **XGBoost** demonstrated the strongest overall performance, achieving the lowest MSE (5030.95) and MAE (46.67), as well as the highest R^2 score (0.4954), indicating a relatively strong fit to the observed data. This suggests that XGBoost effectively captured complex patterns and interactions within the dataset, outperforming both linear and neural network-based approaches.

The **GRU model** also showed competitive performance, with an MSE of 7108.95, MAE of 63.86, and R^2 of 0.287. Although it did not surpass XGBoost, GRU outperformed both Linear Regression and LSTM in all three metrics, confirming its suitability for modeling temporal dependencies in time series data.

Linear Regression, used as a baseline, resulted in an MSE of 8412.01, MAE of 71.10, and an R^2 score of 0.1563. As expected, this model exhibited limited capacity to capture the nonlinear and temporal characteristics of the data.



(a) MAE score

(b) MSE score

(c) R^2 score

Fig. 3: Comparison of evaluation metrics (MAE, MSE, R^2) across four models.

The **LSTM model**, despite its theoretical suitability for time series prediction, underperformed in this context. It produced the highest MSE (9220.01) and MAE (71.35), along with the lowest R^2 score (0.0752). These results suggest potential issues with model configuration, feature selection, or training instability, highlighting the importance of careful hyperparameter tuning when deploying deep recurrent networks.

VII. Discussion

The findings indicate that recurrent neural networks like GRU and LSTM struggled to effectively model flight price data in this study. The simpler GRU model performed poorly, even worse than a naive baseline. The under-parameterized architecture may have failed to capture the complexities of the data. Increasing the model complexity improved performance, as shown by the higher R² and lower error metrics for the deeper GRU model. However, even the more complex GRU only explained about 28.7% of the variance, suggesting that temporal dependencies in the flight price data are either weak, noisy, or overshadowed by static and categorical features.

The LSTM model is theoretically well-suited for time series forecasting due to its ability to learn long-term temporal dependencies. However, its practical performance is highly sensitive to several factors, including the quality of input features, the volume of training data, and the adequacy of hyperparameter tuning. In this study, the LSTM model underperformed relative to all three comparative models—Linear Regression, XGBoost, and GRU—across all evaluation metrics (MSE, MAE, and R²). Notably, performance improved substantially after scaling the target values, increasing the R² score from a negative value to a positive one, indicating that improper scaling may have initially hindered the model's effectiveness.

Two LSTM architectures were compared: a low-parameter model with two layers and a more complex model with four layers and dropout regularization. While the deeper model achieved better results than the simpler variant, it still failed to outperform the other models, as shown in Fig. 4 and Fig. 5. This underperformance may be attributed to several factors, including suboptimal hyperparameter configurations, inadequate input preprocessing, especially time step reshaping, or the noise and variability of the training dataset.

The traditional machine learning model XGBoost, which excels at handling nonlinear relationships in tabular data, showed stronger predictive capabilities. This points to the possibility that static or categorical variables including: route information, booking time, and airline played a more significant role in determining flight prices than time-dependent patterns.

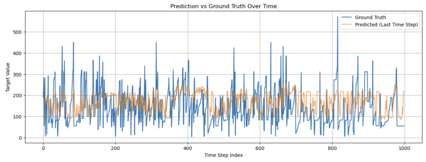


Fig. 4: First 1000 prediction of LSTM model.

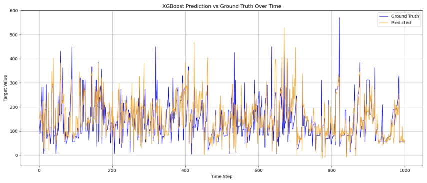


Fig. 5: First 1000 prediction of XGBoost model.

The difficulty in tuning deep learning models further highlights the challenges present in balancing model complexity with generalizability, especially when temporal signals are subtle. Enhancing feature engineering or incorporating external contextual information may be necessary to enhance the sequence-based models in this domain.

In general, these results emphasize the importance of aligning modeling approaches with the underlying structure of the data. Although deep learning approaches hold promise, traditional models remain valuable benchmarks for evaluating and contextualizing the performance of deep learning models.

VIII. Summary

This study examined flight price prediction using a large dataset scraped from Expedia, comparing the performance of baseline models like as Linear Regression and XGBoost to deep learning models like GRU and LSTM. Among these, XGBoost had the highest predictive accuracy, while GRU and LSTM underperformed relative to expectations.

The underperformance of recurrent neural networks indicates that the temporal dynamics in the dataset were either insufficiently strong or too noisy to be properly described without more extensive preprocessing or architectural augmentation.

For future work, we propose several directions to improve prediction performance:

- Enhanced feature engineering to better depict temporal patterns while reducing noise.
- Integration of more external data sources, such as weather conditions, and booking trends, to enrich the dataset.
- Exploration of attention mechanisms and Transformer-based models to model long-range dependencies more effectively.

These findings highlight the need for careful model selection based on the nature of the data, and demonstrate that traditional models remain valuable tools and should not be overlooked.

References

[1] W. Groves and M. Gini, "An agent for optimizing airline ticket purchasing," in *Proceedings of the 2015 International Conference on Autonomous Agents and Multiagent Systems*, International Foundation for Autonomous Agents and Multiagent Systems. AAMAS, 2015, pp. 1461–1469.

[2] A. Joshi, R. Agarwal, and M. Jain, "Skypredict: Airfare prediction machine learning model," *IOSR Journal of Computer Engineering*, vol. 27, no. 2, pp. 35–44, 2022.

[3] W. A. Degife and B.-S. Lin, "Deep-learning-powered gru model for flight ticket fare forecasting," *Applied Sciences*, vol. 13, no. 10, p. 6032, 2023. [Online]. Available: https://www.mdpi.com/2076-3417/13/10/6032

[4] ——, "A multi-aspect informed gru: A hybrid model of flight fare forecasting with sentiment analysis," *Applied Sciences*, vol. 14, no. 10, p. 4221, 2024. [Online]. Available: https://www.mdpi.com/2076-3417/14/10/4221

[5] D. Wong, "Flight prices," https://www.kaggle.com/datasets/dilwong/flightprices, 2025, accessed: 18 May 2025.

[6] W. Luo, "Prediction of flight delays based on the xgboost model," in *2024 International Conference on Computers, Information Processing and Advanced Education (CIPAE)*. Ottawa, ON, Canada: IEEE, 2024, pp. 104–109.

[7] Z. Zhang, Y. Wang, X. Li, Y. Chen, W. Liu, and Y. Sun, "A deep learning approach for flight price prediction," *Applied Sciences*, vol. 13, no. 10, p. 6032, 2023. [Online]. Available: https://www.mdpi.com/2076-3417/13/10/6032