Short-Term High-Resolution Prediction of Airport Baggage Volume Using AI-Based Approaches

1st Necip Gozuacik
Research and Development
Siemens A.S
Istanbul, Turkey
necip.gozuacik@siemens.com

2nd Engin Sag
Research and Development
Siemens A.S
Istanbul, Turkey
engin.sag@siemens.com

3rd Onur Adiguzel
Research and Development
Siemens A.S
Istanbul, Turkey
onur.adiguzel@siemens.com

4th Adem Tekinbas
Research and Development
Siemens A.S
Istanbul, Turkey
adem.tekinbas@siemens.com

5th Sibel Malkos
Research and Development
Siemens A.S
Istanbul, Turkey
sibel.malkos@siemens.com

2025 International Conference on Electrical, Communication and Computer Engineering (ICECCE) | 979-8-3315-4915-2/25/$31.00 ©2025 IEEE | DOI: 10.1109/ICECCE67514.2025.11257932

***Abstract*—The efficient handling of airport baggage is crucial for maintaining smooth airport operations, especially as global air travel rebounds post-pandemic. Despite advancements in AI and predictive analytics, existing studies primarily focus on long-term baggage forecasting, leaving a gap in high-resolution, short-term predictions necessary for real-time operational decisions. This study aims to address this gap by developing a framework for predicting airport baggage volume with 15-minute granularity using AI-based approaches. The research employs a comprehensive methodology involving data collection from multiple airports, feature engineering, and the application of machine learning, deep learning, and time-series models. The dataset, comprising over 4 million records, was processed to extract meaningful features for predictive modeling. Key findings reveal that tree-based ensemble models, particularly the ExtraTreeRegressor, outperform other models, achieving the highest accuracy in predicting baggage volume. These results challenge the assumption that linear models are sufficient for such tasks, highlighting the importance of non-linear methods. The study contributes to the theoretical understanding of short-term baggage prediction by demonstrating the effectiveness of high-resolution temporal features. Practically, it offers airports a robust tool for proactive management, potentially reducing delays and improving passenger satisfaction. The framework's scalability and precision make it an asset for enhancing airport operational efficiency.**

Keywords—airport baggage management, ensemble methods, machine learning, deep learning, temporal analysis

I. INTRODUCTION

The proper handling of baggage is central to the smooth operation of airports. With the global air transport sector rebounding post-COVID-19, the volume and complexity of baggage operations are rising again. According to SITA's 2023 Baggage IT Insights report, global baggage volumes reached nearly 4.35 billion bags in 2022, approaching pre-pandemic levels and expected to grow through 2025 as international travel demand insists on [1].

Passenger expectations have evolved alongside the challenges. The IATA Global Passenger Survey 2023 indicates that 84% of passengers desire real-time baggage updates, and more than half expect minimal baggage waiting times and fewer disruptions during their journeys [2]. To meet such expectations, airports are focusing on predictive analytics and AI-assisted baggage management systems.

While long-term baggage forecasting (daily, weekly monthly) feeds strategic/broader planning, short-term predictions at a high temporal resolution may be crucial/useful for day-to-day operational decisions. Baggage flow can vary significantly over the period of a day or even within minutes, influenced by granular factors such as flight schedules, weather anomalies, and passenger behavior. These minute-level fluctuations mean that real-time or near-real-time forecasts are needed to capture sudden issues. Traditional daily or hourly estimates may not focus/observe over such peaks, whereas a minute-by-minute prediction can reflect them, enabling timely interventions.

Developing AI-driven solutions capable of forecasting baggage counts with fine granularity (hourly or minutely) has therefore become a critical point. Such AI models have the potential to predict baggage loads with high precision. In practice, achieving minute-level prediction accuracy allows airports to react proactively – for example, by opening

979-8-3315-4915-2/25/$31.00 ©2025 IEEE

additional check-in counters or deploying extra baggage carts exactly when they will be needed.

To realize short-term baggage volume prediction, researchers and practitioners have applied a range of AI-based forecasting approaches, including time-series analysis techniques, machine learning models, deep learning networks, and hybrid/ensemble methods. Each approach offers distinct strengths for capturing the patterns and complexities in baggage flow data.

The aim of this study is to propose a framework about short-term prediction of airport baggage volume with using and comparing AI techniques. The novelty here is predicting baggage volume data in 15-minutes granularity such as high resolution focusing.

In Section II, we discuss the available solutions, approaches, and studies in literature. Then, we introduce the details of our proposed framework along with dataset, AI techniques in Section 3. In Section 4, we present our detailed experimental results. Finally, we conclude our paper in Section 5.

II. Related Works

There are several studies in literature regarding prediction of airport baggage volume. both in long-term and short-term granularities. In the following paragraphs, major research outcomes are explained and discussed.

There is a work on check-in baggage flow prediction based on an improved PSO-BP neural network combination model. This study introduces a hybrid PCC-PCA-PSO-BP model that systematically integrates Pearson correlation coefficient analysis, principal component analysis, and PSO-optimized neural networks to enhance predictive performance [3].

Another study provides a detailed assessment of current methodologies and influencing factors in the field [4]. It reveals that although passenger flow is closely related to baggage volume, the relationship is not strictly linear, suggesting that passenger behavior, baggage policies, and seasonal variations play significant roles. The research categorizes influencing factors into macro (e.g., economic indicators, public holidays) and micro (e.g., passenger demographics, flight frequency) levels. From a methodological perspective, it classifies forecasting models into three groups: traditional statistical models, intelligent algorithm-based models, and hybrid models incorporating artificial neural networks.

Managing hand luggage remains an important operational challenge for airlines, contributing to boarding delays and reduced passenger satisfaction. A recent study focusing on KLM Royal Dutch Airlines discovered the underlying reasons of excess hand luggage and proposed a predictive approach using machine learning techniques [5]. The research evaluated several regression models—including Linear Regression, Decision Tree Regression, Random Forest Regression, and XGBoost—on actual flight data, demonstrating that all AI-based models significantly outperformed the existing heuristic used by the airline.

A recent study proposed a forecasting framework to predict recirculating bags (a bag repeatedly loops through parts of the sorting system) [6]. By combining multiple preprocessing steps and implementing the model using Scala and Apache Spark, the solution revealed practical value, achieving a recall of 0.153 and a positive predictive value of 0.764. Beyond prediction, the model also helped to identify features contributing to recirculation events, providing insights for system optimization.

Advanced Air Mobility (AAM) is such an output of advanced technology which needs the development of autonomation of flying and traffic management [7]. Like AAM environments, baggage handling systems require robust, data-driven decision-making in dynamic and high-throughput settings.

III. Methodology

The main motivation of this study is to propose a framework about predicting airport baggage volume from the point of short-term and high-resolution aspects. In this section, we first outline the general pipeline of the system. Subsequently, we describe the dataset, feature engineering and AI models used in the experiments.

Overall system pipeline is displayed in Fig. 1. It consists of three stages: Data Collection, Feature Engineering and AI Development. Each stage also consists of sub-task/steps regarding achievement of the proposed system.

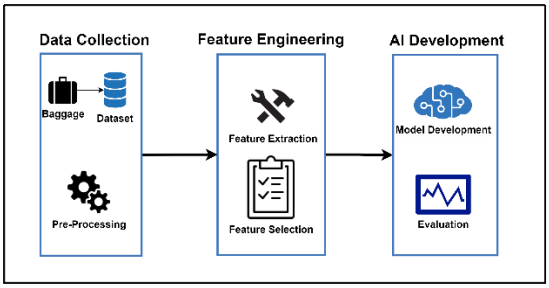

Fig. 1. System pipeline

A. Data Collection

Regarding data collection process, this study examines a baggage dataset where it was acquired from multiple airports. Due to privacy restrictions, the datasets cannot be publicly distributed. The data collection phase spanned eighteen-month period.

The original dataset contained over 20 million records and 25 columns, encompassing comprehensive baggage and flight-related information [8]. Regarding pre-processing, some detailed steps are applied consecutively. After filtering to include only records associated with the designated system airport (in our case, this is Izmir Adnan Menderes Airport), the dataset was reduced to approximately 4.35 million entries. Further refinement to include only flights with a status of "DEPARTED" resulted in a final subset of around 4.11 million records. Key columns in the dataset include flight identifiers (e.g., FLIGHT_CODE, SCHEDULED_TIME), passenger and baggage tracking fields (e.g., PASSENGER_ID, BAG_NUMBER, TAG_NUMBER), and operational indicators such as FLIGHT_STATUS, BAGGAGE_EVENT, and BIM_CREATE_DATE, making it suitable for analyzing baggage handling and flight departure patterns.

For efficient data management, the Cassandra database platform was selected due to its scalability and suitability for handling large volumes of data [9]. Its column-oriented architecture enables high-performance read and write operations across multiple nodes, making it ideal for real-time analytics and large-scale baggage tracking.

B. Feature Engineering

In this study, the focus is to perform short-term prediction with high-resolution such as 15-minutes intervals. Regarding this, it is needed to maintain original data. In the original dataset, there were 25 features. And we applied feature extraction and selection process to transform raw data into meaningful analytical components through several steps.

Firstly, a 15-minute interval-based time windowing approach is applied to a grouped dataset based on SCHEDULED_TIME + FLIGHT_CODE columns/features to facilitate fine-grained temporal analysis. Specifically, for each group, a series of 15-minute intervals in reverse chronological order is generated, starting from the maximum SCHEDULED_TIME and extending back to the minimum BIM_CREATE_DATE (Baggage Record Time). This method enables the segmentation of event data (e.g., baggage processing records) into high-resolution temporal bins, which is particularly valuable for time-sensitive predictive modeling or pattern analysis in domains such as airport operations. After this operation, a new feature called TIME_WINDOW is added into dataset.

Secondly, an aggregation strategy is employed for each distinct TIME_WINDOW sample. In this process, the values from the first record within each window are retained for all features, except for BAG_NUMBER. As BAG_NUMBER serves as the target variable in this study, a different approach is applied: the values are aggregated using a summation method to represent the total number of bags within each corresponding time window. This ensures that the predictive modeling accurately reflects the cumulative baggage volume over the specified intervals.

Lastly, additional time-related features are incorporated to enhance the predictive power of the model. These features are derived from the TIME_WINDOW variable by decomposing its date and time components. For instance, the year is mapped into seasonal categories (Winter, Spring, Summer, Fall), the day is segmented into temporal intervals (Night, Morning, Afternoon), and the hour is classified into operational periods (Morning Peak, Evening Peak, Off-Peak). This temporal enrichment aims to capture cyclical patterns and contextual variations in baggage flow across different periods.

The final representation of the dataset features is summarized in Table I. Among these, 9 features are numerical, while 5 are categorical, reflecting a balanced structure suitable for both statistical and machine learning analyses.

TABLE I. PROPERTIES OF FEATURES

Feature	Type	Description
TIME_MONTH	Numerical	Month (1-12)
TIME_DAY	Numerical	Day of Month (1-31)
TIME_HOUR	Numerical	Hour (0-23)
TIME_MINUTE	Numerical	Minute (0-59)
DAY_OF_YEAR	Numerical	Day of Year
QUARTER	Numerical	Quarter (1-4)
WEEKDAY	Numerical	Day of Week (0-6)
IS_WEEKEND	Numerical	Weekend Information (0-1)
IS_HOLIDAY	Numerical	Holiday Information (0-1)
SEASON	Categorical	Season (winter, spring, summer, fall)
BUSY_PERIOD	Categorical	Peak Stage (morning, evening, off)
CATEGORY	Categorical	Flight Category (Domestic ,International)
TIME_OF_DAY	Categorical	Part of the Day (night, morning, afternoon)

In the context of feature space, the final representation comprises 13 distinct components. Categorical variables are transformed using one-hot encoding to preserve nominal relationships without imposing ordinal assumptions, while numerical features are normalized through feature scaling techniques to ensure comparability and enhance model convergence.

C. AI Development

Employed AI models here are systematically grouped under three main methodological categories: machine learning, deep learning, and time-series modeling. The machine learning category includes ensemble and linear models such as Random Forest, Gradient Boosting, Bagging Regressor, ExtraTree Regressor, Decision Tree etc. These models are good for handling structured tabular data. The deep learning category is represented by the Deep Neural Network (DNN), which excels in modeling high-dimensional data through layered abstractions and non-linear transformations. In the time-series modeling category, the Long Short-Term Memory (LSTM) network is utilized, as it is specifically designed to capture temporal dependencies and sequential patterns within historical baggage data.

One of the proposed AI models is displayed in Fig. 2.for DNN architecture. The model begins with input features and reduces the dimensionality through layers of 64, 32, and 16 neurons, finally producing a single output value, suitable for regression tasks (Target is BAG_NUMBER).

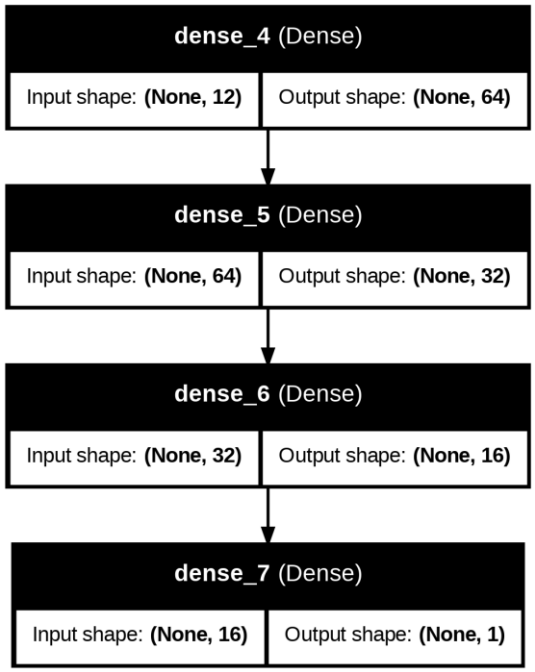

Fig. 2. DNN Architecture

The performance of the proposed model was evaluated using a comprehensive set of metrics to ensure a robust analysis. Mean Absolute Error (MAE) was utilized to measure the average magnitude of prediction errors, offering a straightforward interpretation of model accuracy. Mean Squared Error (MSE) was employed to capture the variability in prediction errors, emphasizing larger deviations. Root Mean Squared Error (RMSE) provided an interpretable measure expressed in the same units as the target variable, facilitating practical understanding. Finally, the coefficient of determination (R^2) was calculated to quantify the proportion of variance in the target variable explained by the model, with values closer to 1 indicating a stronger fit. These metrics collectively offer a multidimensional evaluation of model effectiveness.

IV. Results

The proposed models were ensured through k-fold cross-validation with k = 5, allowing for balanced performance evaluation across different data partitions. Hyperparameter optimization was performed using a systematic grid search strategy [10]. All computational experiments were executed on a high-performance workstation equipped with a dual-GPU setup featuring two NVIDIA GeForce GTX 1080 Ti units (each with 11 GB GDDR5X memory, supporting CUDA version 12.2).

Overall performance comparison is shown in Table II. The regression results indicate that tree-based ensemble models outperform other model families across all evaluation metrics. The ExtraTreeRegressor demonstrates the best overall performance, achieving the lowest RMSE (0.028) and MAE (0.019), alongside the highest R^2 score (0.607). This suggests its strong capability to model complex, non-linear relationships. However, both ExtraTree and Bagging are known to produce very large model sizes, which can be a significant consideration in deployment scenarios where memory or latency constraints exist. Memory consumption or occupied disk size may affect edge deployment in airport terminal systems used for real-time decision-making. Additionally, such ensemble models can bring out substantial inference latency due to the need to traverse deep trees per prediction. This delay may be unacceptable in latency-sensitive operations.

TABLE II. Test set performance

Model	MSE	RMSE	MAE	R2
BaggingRegressor	0.001	0.029	0.020	0.590
DecisionTree	0.001	0.033	0.023	0.476
DNN	0.001	0.032	0.022	0.440
ExtraTreeRegressor	0.001	0.028	0.019	0.607
Gradient Boosting	0.001	0.035	0.025	0.411
HuberRegressor	0.002	0.045	0.032	0.008
LinearRegression	0.002	0.044	0.033	0.051
LSTM	0.001	0.037	0.027	0.262
PoissonRegressor	0.002	0.045	0.034	0.002
RandomForest	0.001	0.031	0.022	0.519
Ridge	0.002	0.044	0.033	0.049
SGDRegressor	0.002	0.044	0.033	0.042

The RandomForest model also yields strong results, with RMSE = 0.031, MAE = 0.022, and R^2 = 0.519, representing a good trade-off between predictive power and model complexity. In contrast, the DecisionTree regressor. Deep learning models show mixed performance: the DNN achieves moderate error rates (RMSE = 0.032, MAE = 0.022, R^2 = 0.440), while the LSTM underperforms significantly (R^2 = 0.262, MAE = 0.027), which may indicate insufficient sequence dependencies in the dataset or suboptimal hyperparameter settings.

Linear and generalized linear models, including LinearRegression, Ridge, SGDRegressor, PoissonRegressor, and HuberRegressor, consistently yield the worst results. This highlights their inability to capture non-linearities present in the data and further reinforces the suitability of non-linear ensemble methods for this task.

To evaluate the contribution of individual features, Shapley Additive Explanations (SHAP) analysis was conducted [11]. The results are visualized in Fig. 3, providing insights into the relative importance and impact of each feature on the model's predictions. Among all features, TIME_HOUR shows the highest influence, followed by TIME_MINUTE, DAY_OF_YEAR, and TIME_OF_DAY, indicating that fine-grained temporal features play a crucial role in the prediction task. In contrast, variables such as IS_HOLIDAY, QUARTER, and TIME_DAY exhibit minimal contribution, as reflected by their narrow SHAP value distributions near zero.

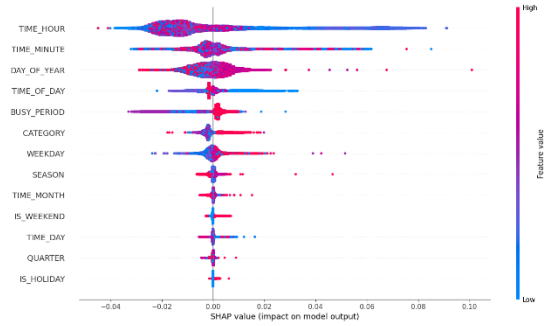

Fig. 3. SHAP Analysis for RandomForest Model

A sample prediction result is visualized in Fig 4. The graphic presents minutely baggage prediction analysis (15-minutes) for Izmir Adnan Menderes (ADB) airport on November 2, 2024, at 3:00 AM (selected date and time), distinguishing between domestic and international flights. It compares the total predicted baggage volume, the average amount of baggage per interval, and the variability in predictions. Domestic flights show significantly higher baggage volumes across all metrics, with 198 bags predicted compared to 71 for international flights.

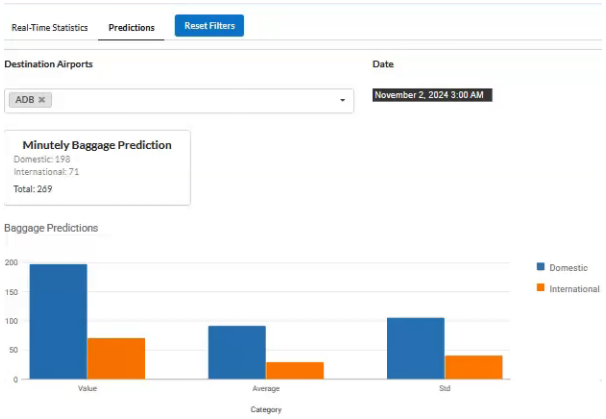

Fig. 4. Prediction Dashboard

The proposed system can be integrated into real-world airport operations through a secure gateway communication infrastructure, including properly designed Application Programming Interfaces (API) for interoperability with existing airport management systems.

V. Conclusion

This study aimed to develop a framework for short-term airport baggage volume prediction with a granularity of 15 minutes, addressing a significant gap in existing literature that primarily focuses on long-term forecasting. The methodology involved data collection from multiple airports, feature engineering (time windowing approach within 15-minutes periods), and the application of various AI techniques, including machine learning and deep learning models. Key findings indicate that tree-based ensemble models, particularly the ExtraTreeRegressor, significantly outperform other approaches, achieving the highest MAE score in predicting baggage volume. This research contributes to the field by demonstrating the effectiveness of high-resolution temporal features in predictive modeling, offering airports a robust tool for proactive management that can enhance operational efficiency and passenger satisfaction.

Despite its contributions, the study has limitations, including the reliance on a single airport dataset, which may affect the generalizability of the findings to other contexts. Future research should explore the application of this framework across diverse airport environments and consider integrating additional datasets to enhance model robustness. Additionally, investigating alternative AI methodologies, such as hybrid models that combine different approaches, could yield further insights into baggage volume prediction. The practical implications of this research are significant, as improved predictive capabilities can lead to better resource allocation and reduced delays in airport operations.

Acknowledgment

This study was supported by Eureka-ITEA Project "SOCFAI" (Project Number: ITEA-21020). We extend our gratitude to TUBITAK for funding this project. Our special thanks go to our project partners, TAV Technologies, Siemens A.S for their invaluable contributions and collaboration.

References

[1] SITA, Baggage IT Insights 2023. [Online]. Available: https://www.sita.aero/resources/surveys-reports/baggage-it-insights-2023/. [Accessed: July 9, 2025].

[2] International Air Transport Association, IATA Global Passenger Survey Highlights, 2019. [Online]. Available: https://www.iata.org/en/publications. [Accessed: July 9, 2025].

[3] B. Jiang, J. Zhang, J. Fu, G. Ding, and Y. Zhang, "Research on check-in baggage flow prediction for airport departure passengers based on improved PSO-BP neural network combination model," Aerospace, vol. 11, no. 11, p. 953, 2024.

[4] B. Jiang, G. Ding, J. Fu, J. Zhang, and Y. Zhang, "An overview of demand analysis and forecasting algorithms for the flow of checked baggage among departing passengers," Algorithms, vol. 17, no. 5, p. 173, 2024.

[5] S. Boot, Predicting the amount of excess hand luggage on an aircraft using machine learning, M.S. thesis, 2024.

[6] G. M. van der Sanden, A Forecasting Framework for Recirculation in Baggage Handling Systems, M.S. thesis, Eindhoven Univ. of Technology, 2020.

[7] S. Bae and J. Kim, "Exploring dialogue capabilities with automation systems in the AAM ecosystem," in Proc. 2024 Int. Conf. Electr., Commun. Comput. Eng. (ICECCE), 2024, pp. 1–7.

[8] N. Gozuacik, A. Tekinbas, E. Sag, O. Adiguzel, and S. Malkos, "Temporal pattern analysis of baggage impact on flight operations," in Proc. 14th Int. Conf. Pattern Recognit. Appl. Methods (ICPRAM), 2025, pp. 831–838. doi: 10.5220/0013372800003905.

[9] H. Tu, "Cassandra vs. MongoDB: A systematic review of two NoSQL data stores in their industry uses," in Proc. 2024 IEEE 7th Int. Conf. Big Data Artif. Intell. (BDAI), 2024, pp. 81–86.

[10] T. Agrawal, Hyperparameter optimization using scikit-learn, in Hyperparameter Optimization in Machine Learning: Make Your Machine Learning and Deep Learning Models More Efficient, Springer, 2021, pp. 31–51.

[11] Y. Nohara, K. Matsumoto, H. Soejima, and N. Nakashima, "Explanation of machine learning models using improved shapley additive explanation," in Proc. 10th ACM Int. Conf. Bioinformatics, Comput. Biol. Health Informat., 2019, pp. 546–546

A Multimodal LLM-Based AI Assistant for Intelligent Exploration of Airport Baggage Records

2026 5th International Informatics and Software Engineering Conference (IISEC) | 979-8-3315-8031-5/26/$31.00 ©2026 IEEE | DOI: 10.1109/IISEC69317.2026.11418498

Onur Adiguzel
Research and Development
Siemens A.S
Istanbul, Turkey
onur.adiguzel@siemens.com

Necip Gozuacik
Research and Development
Siemens A.S
Istanbul, Turkey
necip.gozuacik@siemens.com

Adem Tekinbas
Research and Development
Siemens A.S
Istanbul, Turkey
adem.tekinbas@siemens.com

Engin Sag
Research and Development
Siemens A.S
Istanbul, Turkey
engin.sag@siemens.com

Sibel Malkos
Research and Development
Siemens A.S
Istanbul, Turkey
sibel.malkos@siemens.com

***Abstract*—The field of airport operations is increasingly reliant on advanced data analytics to enhance efficiency and passenger satisfaction, particularly in the management of baggage handling systems. Despite significant advancements in artificial intelligence, existing literature reveals a gap in the application of large language models (LLMs) for interactive exploration of baggage data, limiting decision-makers' ability to derive actionable insights from complex datasets. This study aims to develop a multimodal LLM-based AI assistant that facilitates intuitive interaction with large-scale airport baggage records, addressing the identified need for more accessible and interpretable data analysis tools. Utilizing containerized microservice architecture, the research employs a dataset comprising four million baggage entries from Izmir Adnan Menderes Airport, integrating various open-source LLMs for natural language and voice input processing. Key findings indicate that the Qwen3-30B-A3B-Instruct model outperforms others in processing speed and accuracy in service redirection, demonstrating the potential of LLMs to enhance operational analytics. This work contributes to the theoretical understanding of multimodal AI applications in airport management and offers practical implications for improving data-driven decision-making processes, ultimately supporting more efficient and transparent airport operations. The proposed architecture not only advances the capabilities of AI assistants in complex environments but also lays the groundwork for future innovations in digitalization strategies within the aviation sector.**

Keywords—, artificial intelligence, baggage analytics, chatbot, data-driven airport operations, large language model, natural language processing

I. Introduction

Airports are complex, data-intensive environments where operational efficiency depends on the coordinated interaction of multiple subsystems, including flight operations, ground services, passenger flow, and baggage handling. Among these, baggage operations play a critical role in ensuring on-time departures and maintaining passenger satisfaction, yet they remain one of the least transparent and most challenging components to manage [1]. Modern baggage handling systems generate massive volumes of heterogeneous data, including structured event records, time-stamped operational logs, and contextual metadata, making manual analysis increasingly impractical.

Airport operators increasingly face the need for interactive and explainable access to operational data. Decision-makers often need to ask ad-hoc questions, investigate unexpected behaviors, compare historical patterns, or understand the reasoning behind Artificial Intelligence (AI)-generated predictions. Traditional query-based systems and visualization tools are insufficient for this purpose, as they demand prior knowledge of data schemas, metrics, and analytical pipelines. This gap highlights the need for intelligent interfaces that can translate complex operational data into human-centric, interpretable insights.

Recent advances in LLMs offer a promising opportunity to address this challenge. LLMs have demonstrated strong capabilities in natural language understanding, contextual reasoning, and semantic knowledge integration. When combined with enterprise data sources, retrieval mechanisms, and domain-specific constraints, LLMs can act as cognitive intermediaries between humans and complex data ecosystems. Furthermore, the integration of multimodal components—such as natural language queries, temporal summaries, and visual analytics—enables richer interaction paradigms beyond traditional text-based querying.

Despite their rapid adoption in general-purpose applications, the use of LLM-based AI assistants in mission-critical airport operations, particularly baggage data exploration, remains largely unexplored. Key challenges include handling large-scale structured time-series data, ensuring reliable and explainable responses, and integrating predictive analytics with interactive reasoning. Addressing these challenges requires system-level designs that go beyond standalone models and focus on end-to-end operational usability.

In this paper, we propose a multimodal LLM-based AI assistant for intelligent exploration of airport baggage records. The proposed system enables users to interact with large-scale baggage datasets through natural language, while seamlessly integrating historical records, patterns, predictions and visualizing graphics automatically. The contributions of this study are the following:

979-8-3315-8031-5/26/$31.00 ©2026 IEEE

- To integrate open-source LLMs with large-volume airport baggage records by extracting correct parameters and mapping to the proper services.
- To propose multimodal service with accepting both text and voice input from users.
- To provide and visualize baggage analytics, baggage predictions based on pre-defined services.

II. Related Work

AI–generated content highlights a growing shift toward agent-based frameworks to overcome the limitations of standalone LLMs in complex industrial applications. In particular, collaborative multi-agent systems may address challenges related to domain-specific semantic understanding, multi-round reasoning, and decision-making by enabling feedback-driven interaction among specialized agents [2].

Studies highlight that human–machine collaboration typically follows a structured workflow involving task definition, division of responsibilities, iterative interaction, and decision execution, with natural language processing and multimodal learning serving as core enabling technologies. These approaches have demonstrated applicability across diverse domains, including business intelligence, smart manufacturing, autonomous driving, and medical diagnosis, while future research increasingly points toward expert large models, embodied intelligent agents, and brain–computer interfaces as key directions for advancing collaborative AI systems [3].

From an AI analysis perspective, recent literature increasingly emphasizes the role of multimodal intelligence in advancing airport baggage security systems. In this context, the STING-BEE study represents a significant contribution by proposing a multimodal AI assistant that integrates visual X-ray imagery with textual representations to support more intelligent and explainable security inspection [4].

In smart manufacturing environments, heterogeneous data sources such as visual inspection images, sensor measurements, and production records must be jointly analyzed, which exposes the limitations of traditional single-modal approaches. Multimodal LLM–based systems address these challenges by providing unified data representations, dynamic semantic tokenization, and strong cross-modal alignment mechanisms [5].

Conversational AI in air transportation highlights the growing use of LLMs to enhance customer interaction and decision support during flight booking processes. Domain-specific LLMs integrated with Retrieval-Augmented Generation (RAG) frameworks have emerged as an effective solution by combining natural language understanding with structured aviation knowledge bases and external Application Programming Interfaces (APIs). Chatbot-based systems can support end-to-end flight booking workflows, leveraging vector databases for semantic retrieval and transactional databases for user and booking management [6].

In a study, the use of ChatGPT is investigated as a domain-specific conversational AI for the tourism industry, focusing on Setur's products and services. It examines intention recognition and response generation by integrating company-specific data such as hotel information and catalogs. Rather than relying on a fully autonomous LLM, the work evaluates multiple architectural designs and proposes a hybrid framework that combines LLM-based dialogue with intent-based agent selection and traditional retrieval methods [7].

III. Methodology

This study plans to develop an assistant powered by LLM, including a multimodal approach as of text and voice. In this section, we first outline the general architecture of the system and describe the AI-Assistant implementation in detail.

The presented architecture in Fig. 1 illustrates a containerized, microservice-based AI analytics platform in which all major system functionalities are deployed as independent Docker containers to ensure modularity, scalability, and fault isolation. The AI Assistant container serves as the primary interaction layer, enabling users to access the system through natural language queries and coordinating requests across backend services, while the Dashboard container provides a visual interface for monitoring analytical results and system outputs. Management container handles the incoming service requests and routes to the corresponding service. Data analytical operations are performed within a Data Analysis container, which prepares proper Structured Query Language (SQL) queries and retrieves from Database. To support forecasting at different temporal granularities, the architecture includes separate Minutely, Hourly, and Daily Prediction containers, each encapsulating models optimized for its respective time horizon. Additionally, working principle of Minutely Prediction service was also described with details in [8]. Overall, this loosely coupled, service-oriented design facilitates efficient deployment, independent service evolution, and seamless integration of AI-driven analytics and decision-support capabilities within a unified platform.

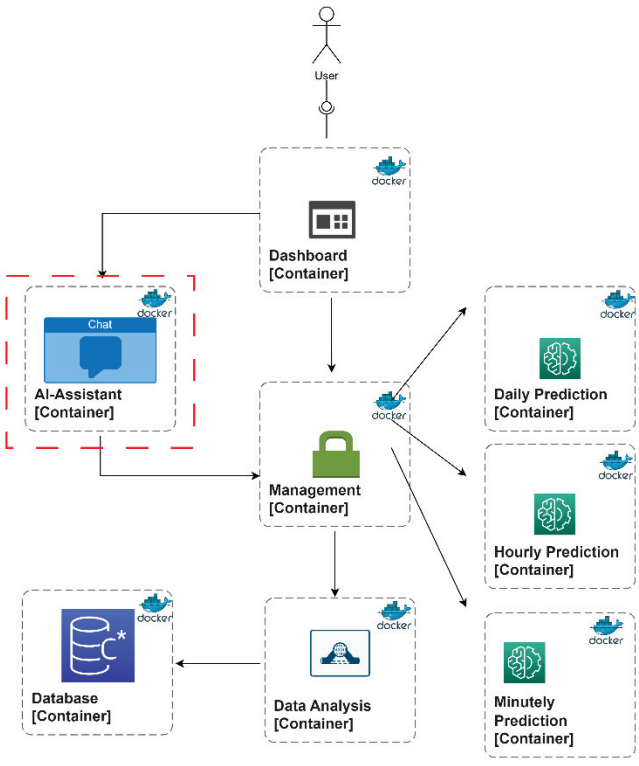

Fig. 1. System Architecture

We developed a modular, service-oriented backend architecture regarding AI-Assistant component that enables natural language interaction with airport baggage analytics and prediction systems as shown in Fig. 2. The primary objective of the system is to bridge unstructured user queries and structured backend services by leveraging LLMs as an intelligent interpretation and orchestration layer. The implementation adopts a Flask-based RESTful interface and integrates multiple analytical and predictive microservices, allowing users to query complex operational data through conversational inputs in both textual and spoken form.

At the core of the architecture the central orchestration and intelligence component is responsible for configuring and managing interactions with the LLM endpoint, including model selection, authentication, timeout handling, and request formatting. In addition, it initializes and coordinates several auxiliary modules, such as the query transformer, date parser, backend function mapper, response processor, and tool selector. Together, these components form a structured processing pipeline that transforms natural language input into machine-interpretable representations and subsequently into executable backend requests.

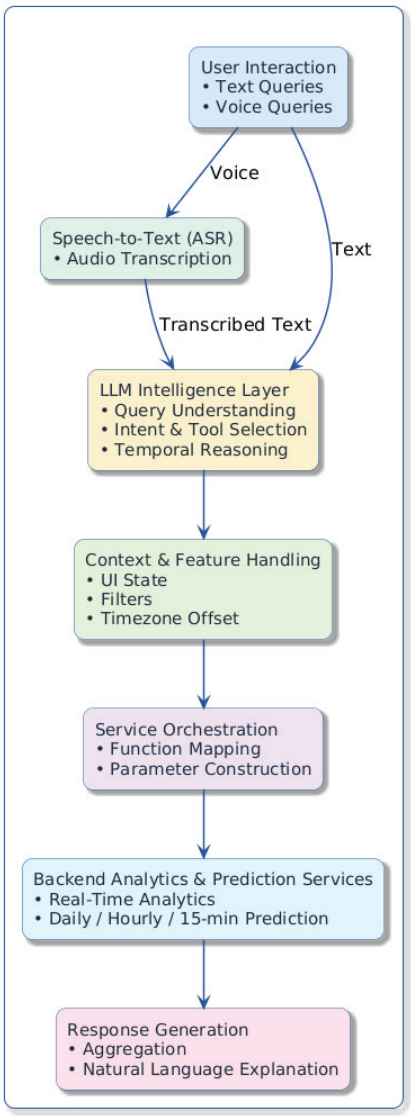

Fig. 2. AI-Assistant Pipeline

The query processing workflow begins with contextual interpretation, where the system extracts temporal information from the user interface state to correctly resolve relative and absolute time expressions (local time zone and UTC). The natural language query is then processed and then converted into a structured semantic representation capturing the query type, temporal scope, operational category, and domain-specific entities such as airports or flight attributes. This intermediate representation enables deterministic downstream processing while preserving the semantic intent of the original query.

Although LLMs possess a wealth of aviation-related information, they do not inherently understand our specific business logic and data structures. Therefore, the query understanding process involves sophisticated prompt engineering to convert user input into the precise parameters required by our system. To ensure accuracy, data and project-specific details are explicitly provided in the prompt, as LLM may not possess this information or might interpret it differently. Additionally, one-shot and few-shot prompting techniques are employed to facilitate fine-tuning and enhance the system's overall robustness.

Following query understanding, the system employs an LLM-driven tool selection mechanism to determine the most appropriate backend service for fulfilling the request. Depending on the identified intent, the query is routed either to real-time data analysis services or to predictive services operating at different temporal granularities (daily, hourly, or minutely). The semantic query representation is translated into concrete REST API parameters, ensuring compatibility with backend service contracts and encapsulating domain-specific mapping logic.

The LLM Intelligence Layer here is designed to work in conjunction with existing baggage analytics, mapping the parameters to the pre-defined SQL queries. It is not allowed to generate SQL queries or connect to the underlying database to perform the relevant calculations. The motivation here is to prevent LLMs from making mistakes even in simple multiplication/division operations, as well as to avoid potentially incorrectly written SQL queries. In this way, we let the user unlimited query possibilities see accurate results for all available metrics on the User Interface (UI) (for example, for any desired date/destination/airline, etc.). This design decision ensures both computational accuracy and system reliability by separating natural language understanding from direct database manipulation, while maintaining the flexibility and expressiveness of conversational interfaces.

Data retrieval is performed through synchronous HTTP requests to the selected backend microservices. The system incorporates explicit handling of temporal normalization and time zone offsets to maintain consistency between user queries and backend data representations. The response is interpreted from the raw data and frontend interface generates a natural language summary that aligns with the original user intent, thereby completing the end-to-end transformation from conversational input to analytical insight.

Lastly, user inputs can be categorized into short and long queries based on utterance length and semantic complexity, independent of input modality. Short queries correspond to brief text or spoken utterances with limited semantic scope, typically involving a single temporal constraint, whereas long queries represent extended expressions that combine multiple

parameters such as airline, flight type, airport locations, and time constraints.

IV. Experimental Results

The proposed architecture is examined and utilized within the prepared dataset collected from Izmir Adnan Menderes Airport which consists of 4 million entries [9]. Implemented Dockerized AI-Assistant service is enabled and integrated into the already working platform given in Fig. 1.

Regarding natural language input, three open-source LLMs are identified to be used. These are mistral-7b-instruct, qwen3-30b-a3b-instruct-2507 and gpt-oss-120b. Mistral-7B-Instruct is an instruction-tuned LLM developed by Mistral AI that's designed to follow user instruction effectively while being efficient in compute and capable of strong performance across many tasks [10]. Qwen3-30B-A3B-Instruct-2507 is a 30 billion parameter instruction-tuned LLM based on Mixture-of-Experts (MoE) architecture. It's part of the Qwen3 series, a next-generation family of LLMs designed to deliver strong performance across language understanding, generation, reasoning, coding, and multilingual tasks [11]. GPT-OSS-120B is an open-weight LLM released by OpenAI in 2025, marking one of the company's first major publicly available model releases since GPT-2 [12]. Regarding the handling of voice/speech input, Whisper Large V3 Turbo model is applied [13]. It is an Automatic Speech Recognition (ASR) model developed by OpenAI for high-speed transcription with minimal quality loss.

The open-source LLMs are evaluated based on processing time, correct tool selection and parameter extraction. Processing time which is measured in seconds directly impacts system efficiency because longer response times increase operator wait. Correct tool selection is the most critical one therefore LLMs must accurately identify the appropriate system tool (as shown in Fig. 1.) for user prompts, whether for historical data analysis or multi-granularity predictions. Parameter extraction is equally critical, since it determines query parameters for these tasks, necessitating accurate extraction of all user-entered values.

For testing, two distinct queries were employed: a "short query" like ("total number of bags on 17 January 2024?") with only a date parameter, and a "long query" ("total number of bags for departed flights of PC airline in ADB and SAW airports on 17 January 2024?") including five parameters (airline, flight type, two airports, and date). The results for processing time, output token count and correct tool selection are presented in Table I. On the other hand, all models successfully extracted every parameter correctly during the parameter extraction test.

TABLE I. PERFORMANCE COMPARISON FOR TEXT INPUT

Model	Input Type	Processing Time (s)	Output Token Count	Redirection to Correct Service
mistral-7b-instruct	Short Query	7.23	268	0%
	Long Query	7.45	281	100%
qwen3-30b-a3b-instruct-2507	Short Query	3.32	551	100%
	Long Query	3.74	620	100%
gpt-oss-120b	Short Query	9.62	492	100%
	Long Query	10.76	529	100%

The mistral-7b-instruct model exhibits relatively high processing latency for both short and long queries (7.23 s and 7.45 s, respectively). While it completely fails to redirect short queries to the correct service (0%), it achieves perfect accuracy for long queries (100%). In contrast, qwen3-30b-a3b-instruct-2507 demonstrates consistently superior performance across all evaluated dimensions. It achieves the lowest processing times (3.32 s for short queries and 3.74 s for long queries) while maintaining 100% redirection accuracy for both query types. These results indicate strong instruction-following capabilities and efficient inference, likely benefiting from its MoE architecture and optimized routing behavior. The gpt-oss-120b model achieves perfect redirection accuracy (100%) for both short and long queries, confirming its strong reasoning and intent classification capabilities. However, this accuracy comes at the cost of significantly higher processing latency (9.62 s and 10.76 s). Such latency may limit its applicability in real-time or low-latency service routing scenarios, despite its robustness. From the view of output token count, qwen3-30b-a3b-instruct-2507 produces the highest number of tokens; however, in terms of processing time efficiency, it clearly outperforms the other models, indicating superior inference and runtime optimization.

These findings suggest that model scale alone does not guarantee optimal system-level performance. Instead, architectural efficiency and instruction tuning play a decisive role in achieving both fast and accurate service redirection. For production-grade, real-time systems, Qwen3-30B-A3B-Instruct-2507 emerges as the most suitable candidate among the evaluated models. Additionally, a sample interaction with AI-Assistant is displayed in Fig. 3. The user provides the query from the chatbot area, and the response is shown there after all the background operations are completed.

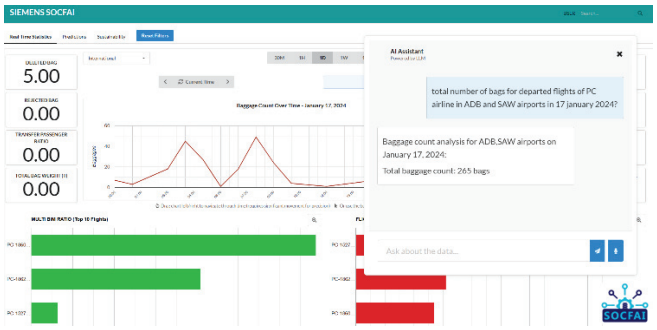

Fig. 3. Dashboard View of AI-Assistant

Regarding voice/speech input, user interaction was evaluated using a single ASR model, namely Whisper Large V3 Turbo, integrated into the multimodal service routing pipeline. Although no inter-model comparison was conducted, the evaluation focuses on performance robustness under varying input conditions and system-level effectiveness. The results show that the model achieves high transcription reliability across both short and long voice

queries. For short utterances, the average transcription latency remained low, supporting near real-time interaction. For longer speech inputs, processing time increased moderately but remained within acceptable bounds for interactive systems.

Despite these results, there are also limitations that need to be acknowledged. First, more complex queries and stress testing should be considered from the point of accessibility of the service. Second, voice input was assessed using a single ASR model, preventing a comparative analysis across alternative speech models or noisy operational settings.

Beyond the quantitative performance metrics presented, our experimental framework and architectural design also aim to evaluate considerations for real-world deployment, specifically concerning applicability across diverse/multiple airport environments, data security, and regulatory compliance. By abstracting the natural language interface from specific backend data schemas and integrating via standardized APIs, the system is designed for configurability, enabling its potential deployment across multiple airports with differing baggage handling systems through appropriate service mapping and data ingestion strategies. Furthermore, the experiments confirm the system's ability to reliably translate conversational queries with guidance of system prompts into structured service calls without allowing direct database manipulation by the LLM. This fundamental design choice, where the LLM acts as an intelligent interpreter and orchestrator of pre-defined, secure operations, is paramount for ensuring data security, maintaining stringent access control, and facilitating compliance with aviation data regulations, as it prevents unauthorized data access or modification.

V. Conclusion

This study aimed to design and evaluate a multimodal LLM-based AI assistant that enables intuitive exploration of large-scale airport baggage records through natural language and voice interaction. The proposed solution adopts a containerized, microservice-oriented architecture in which an LLM functions as an orchestration layer that interprets user intent, selects appropriate analytical or predictive services, and maps extracted parameters to predefined backend queries. A key insight here is the proper effectiveness of an LLM as an intelligent orchestration layer. Experimental evaluation on a real-world dataset comprising approximately four million baggage records demonstrates that the system can reliably translate conversational queries into structured service calls without allowing direct database manipulation. The results show that model choice significantly affects response latency and service routing accuracy, with Qwen3-30B-A3B-Instruct achieving the most balanced performance. Overall, the study contributes a practical system architecture and empirical evidence showing that multimodal LLM-driven interfaces can improve accessibility, reliability, and interpretability in operational airport analytics while preserving computational correctness. Additionally, this approach opens new horizons for investigating human-AI collaboration in critical operational decision-making procedures.

Future work can include cross-lingual evaluations, and systematic benchmarking of speech recognition components under varying acoustic conditions. The integration of the Model Context Protocol (MCP) can be evaluated as a structural and architectural enhancement to provide a standardized mechanism for exposing analytical and predictive services as formally defined tools, enabling deterministic tool invocation and consistent context exchange between the LLM and backend services. Integration of text-to-speech can also be studied to increase the usability of the system. From a practical perspective, the proposed architecture has the potential to reduce training effort for decision-makers and to support more transparent, data-driven airport operations, which may inform future digitalization strategies in air transportation management.

Acknowledgment

This study was supported by Eureka-ITEA Project "SOCFAI" (Project Number: ITEA-21020). We extend our gratitude to TUBITAK for funding this project.

References

[1] L. Ranasinghe, “Analyzing the determinants influencing the customer satisfaction level of Bandaranaike International Airport services (BIA): A quantitative approach,” 2025.

[2] H. Chen and F. Pang, “A decision-support agent framework and its application in industry,” in Proceedings of the 2024 3rd International Conference on Cyber Security, Artificial Intelligence and Digital Economy, 2024, pp. 131–137.

[3] Z. Liu and Y. Peng, “Human–machine interactive collaborative decision-making based on large model technology: Application scenarios and future developments,” in Proceedings of the 2nd International Conference on Artificial Intelligence and Digital Technology (ICAIDT), IEEE, 2025, pp. 106–110.

[4] D. Velayudhan, A. Ahmed, M. Alansari, N. Gour, A. Behouch, T. Hassan, S. T. Wasim, N. Maalej, M. Naseer, J. Gall, et al., “STING-BEE: Towards vision-language model for real-world X-ray baggage security inspection,” in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2025, pp. 20767–20777.

[5] T. Wang, B. Zhang, D. Jiang, and D. Li, “A multimodal large language model framework for intelligent perception and decision-making in smart manufacturing,” Sensors, vol. 25, no. 10, p. 3072, 2025.

[6] M. Raj, Y. S. Kishore, and M. G. Jenifel, “LLM for airline assistance using retrieval augmented generation and MongoDB,” in Proceedings of the International Conference on Innovations and Advances in Cognitive Systems, 2025, pp. 291–303.

[7] O. M. Kılıçlıoğlu, Ş. T. Özçelik, and M. T. Yöndem, “Application of ChatGPT in the tourism domain: Potential structures and challenges,” in Proceedings of the 4th International Informatics and Software Engineering Conference (IISEC), IEEE, 2023, pp. 1–4.

[8] N. Gözüaçık, E. Sağ, O. Adıgüzel, A. Tekinbaş, and S. Malkoş, “Short-Term High-Resolution Prediction of Airport Baggage Volume Using AI-Based Approaches,” in Proc. 2025 Int. Conf. Electrical, Communication and Computer Engineering (ICECCE), 2025, pp. 1–5, doi: 10.1109/ICECCE67514.2025.11257932.

[9] N. Gozuacik, A. Tekinbas, E. Sag, O. Adiguzel, and S. Malkos, “Temporal pattern analysis of baggage impact on flight operations,” in Proc. 14th Int. Conf. Pattern Recognit. Appl. Methods (ICPRAM), 2025, pp. 831–838. doi: 10.5220/0013372800003905.

[10] .Q. Jiang et al., “Mistral 7B,” arXiv preprint, 2023.

[11] A. Yang, A. Li, B. Yang, B. Zhang, B. Hui, B. Zheng, B. Yu, C. Gao, C. Huang, C. Lv, et al., “Qwen3 technical report,” arXiv preprint, arXiv:2505.09388, 2025.

[12] S. Agarwal, L. Ahmad, J. Ai, S. Altman, A. Applebaum, E. Arbus, R. K. Arora, Y. Bai, B. Baker, H. Bao, et al., “gpt-oss-120b & gpt-oss-20b model card,” arXiv preprint, arXiv:2508.10925, 2025.

[13] Fireworks, “Whisper-large-v3-turbo,” 2025. [Online]. Available: https://fireworks.ai/models/fireworks/whisper-v3-turbo

Predictive Modeling of Airport Flight Delays Using AI Techniques

Araştırma Makalesi/Research Article

Adem TEKİNBAS[1], Burak KADIOGLU[2], Necip GOZUACIK[1]*, M. Gozde SAYIN[2], Engin SAG[1], Onur ADIGUZEL[1], Sibel MALKOS[1]

[1]Research and Development, Siemens A.S, Istanbul, Turkey
[2]Research and Development, TAV Technologies, Istanbul, Turkey

adem.tekinbas@siemens.com, consultant.burak.kadioglu@tav.aero, necip.gozuacik@siemens.com, gozde.sayin@tav.aero, engin.sag@siemens.com, onur.adiguzel@siemens.com, sibel.malkos@siemens.com

(Geliş/Received:19.08.2025; Kabul/Accepted:03.12.2025)
DOI: 10.17671/gazibtd.1768483

***Abstract*—** Air transport is a critical component of global economy, facilitating business, tourism, and logistics. However, flight delays pose significant challenges, leading to economic losses and passenger dissatisfaction. Recent studies employ machine/deep learning for delay prediction but largely rely on single-source datasets such as schedules and weather, treat arrival and departure delays independently, and rarely integrate operational processes like baggage handling. This study introduces a holistic multi-source data integration framework for predictive modeling of airport flight delays using advanced Artificial Intelligence (AI) techniques. The framework combines flight schedules, Automatic Dependent Surveillance–Broadcast (ADS-B) data, weather conditions, airport-specific features, and baggage-handling information to build comprehensive predictive models for arrival and departure delays. Our primary contribution is a novel integration methodology and the introduction of the "Dissimilarity Ratio," a baggage-derived feature that enhances prediction accuracy. For arrival delay prediction, Random Forest demonstrates superior short-term performance, achieving Mean Absolute Error (MAE) of 2.23 minutes. For departure delay prediction, XGBoost is the optimal model, with baggage-specific features improving Mean Squared Error (MSE) by up to 24.97%. These results show measurable improvements over prior studies by leveraging multi-layered operational data within a unified predictive framework and offer practical implications for improving airport efficiency and passenger satisfaction.

***Keywords*—** aviation management, deep learning, flight delay prediction, machine learning, multi-source data integration, predictive analytics

Havalimanı Uçuş Gecikmelerinin Yapay Zeka Teknikleriyle Öngörüsel Modellenmesi

***Özet*—** Havayolu taşımacılığı, iş dünyası, turizm ve lojistik faaliyetlerini destekleyerek küresel ekonominin kritik bir bileşeni haline gelmektedir; ancak uçuş gecikmeleri ekonomik kayıplara ve yolcu memnuniyetsizliğine yol açan önemli operasyonel sorunlar oluşturmaktadır. Son yıllarda yapılan çalışmalar, gecikme tahmini için makine öğrenmesi ve derin öğrenme yöntemlerini kullanmakla birlikte çoğunlukla uçuş programları ve hava durumu gibi tek kaynaklı veri setlerine dayanmakta, varış ve kalkış gecikmelerini birbirinden bağımsız ele almakta ve bagaj işleme süreçleri gibi operasyonel unsurları sınırlı biçimde bütünleştirmektedir. Bu çalışma, havaalanı uçuş gecikmelerinin kestirimi için gelişmiş Yapay Zekâ tekniklerini kullanan bütüncül ve çok kaynaklı bir veri entegrasyon çerçevesi sunmakta olup uçuş programları, Automatic Dependent Surveillance–Broadcast (Otomatik Bağımlı Gözetim – Yayın) verileri, hava durumu koşulları, havaalanına özgü özellikler ve bagaj işleme bilgilerinin bir araya getirilmesiyle hem varış hem de kalkış gecikmeleri için kapsamlı tahmin modelleri oluşturulmuştur. Çalışmanın temel katkısı, yenilikçi bir veri entegrasyon metodolojisinin geliştirilmesi ve tahmin performansını artıran bagaj temelli yeni bir özellik olan Benzerlik Dışı Oranı'nın (Dissimilarity Ratio) literatüre kazandırılmasıdır. Varış gecikmesi tahmininde Random Forest algoritması kısa vadeli kestirimlerde üstün performans göstererek Ortalama Mutlak Hata (Mean Absolute Error, MAE) değerinde 2.23 dakika elde etmiş; kalkış gecikmesi tahmininde ise XGBoost modeli en yüksek başarıyı göstermiş ve bagajla ilişkili özelliklerin kullanılması Ortalama Kare Hata (Mean Squared Error, MSE) değerinde %24.97'ye varan iyileşme sağlamıştır. Bu bulgular, çok katmanlı operasyonel verilerin bütünleşik bir kestirim çerçevesinde kullanılmasının önceki çalışmalara kıyasla ölçülebilir performans iyileştirmeleri sağladığını ortaya koymakta ve havaalanı operasyon verimliliğinin artırılması ile yolcu memnuniyetinin geliştirilmesine yönelik pratik çıkarımlar sunmaktadır.

***Anahtar Kelimeler*—** havacılık yönetimi, derin öğrenme, uçuş gecikmesi tahmini, makine öğrenimi, çok kaynaklı veri entegrasyonu, öngörüsel analizler

1. INTRODUCTION

In today's fast-paced global economy, air transport plays an indispensable role in facilitating business, tourism, and logistics. Effective airline schedule planning is essential for carriers to achieve operational and financial goals. However, these schedules are typically developed under the assumption of ideal, disruption-free conditions. In practice, numerous factors, such as adverse weather, mechanical failures, congestion in air traffic, and security concerns, often cause delays and operational disruptions [1].

One of the major challenges in aviation management is flight delays, which not only disrupt airline operations and passenger schedules but also lead to significant economic and environmental costs. According to Federal Aviation Administration (FAA) [2] estimates, delays resulted in $33 billion in costs and lost time for United States (US) airspace users and consumers in 2019. In another report prepared by EUROCONTROL [3], nearly 50% of delays within the European air network are classified as 'reactionary delays', while delays attributed to 'airline-related causes' constitute the second most frequent category. Airports in the US are more frequently impacted by convective storms, resulting in significant operational disruptions, whereas European airports are predominantly affected by low-visibility conditions, which have a greater influence on their performance indicators [4].

Short-term flight delay prediction outlines a critical operational capability for airport management system, with implications that extend far beyond simple schedule adherence. When airports can accurately forecast delays within minutes granularity rather than hours, they unlock operational advantages across multiple interconnected systems. For example, gate assignments can be dynamically optimized to maximize terminal capacity utilization, while ground handling resources—from baggage systems can be allocated with just-in-time precision. Fuel consumption is reduced through better engine start timing and taxi planning, while crew scheduling becomes more resilient against regulatory duty-time limitations. Additionally, passenger connection management improves when airlines can make informed decisions about whether to hold connecting flights based on high-confidence arrival predictions

The causes of flight delays are multi-factorial and highly variable across regions and time frames. These include weather anomalies, airspace and airport congestion, late-arriving aircraft, maintenance issues, crew scheduling constraints, and regulatory interventions [5]. Traditional statistical models, such as regression analysis and time series forecasting, have long been employed to understand and mitigate these issues. However, the inherent non-linearity, temporal complexity, and feature interdependence in flight operations demand more adaptive and intelligent approaches.

Recent advances in AI, Machine Learning and Deep Learning offer significant potential in modeling and forecasting flight delays with improved precision [6-8]. Supervised learning techniques such as decision trees, Gradient Boosting Machines (GBMs) , and deep neural networks have been successfully applied to predict both the probability of delay (classification) and estimated delay time (regression). While existing research has made significant strides in flight delay prediction, several critical gaps remain. Most early studies rely on single-source datasets, typically combining publicly available flight records with weather data. In contrast, multi-source approaches have emerged to address the complex dependencies influencing flight delays. A few recent efforts aim to bridge arrival–departure interdependencies, but most still treat these as separate problems, missing the potential operational insights that arise from joint modeling of both perspectives. Furthermore, despite growing interest in comprehensive operational analytics, baggage handling processes—a critical factor influencing both departure punctuality and turnaround time—remain largely underexplored in existing literature.

Moreover, Explainable AI (xAI) has emerged as a critical advancement, allowing aviation stakeholders to understand which features (e.g., wind speed, turnaround time, airport congestion) most strongly influence delay predictions [9-11]. Tools such as Shapley Additive Explanations (SHAP) and Sobol sensitivity analysis help decode the "black box" nature of complex machine learning models, supporting trust, transparency, and action-ability in high-stakes aviation environments.

Given these developments, research contribution and novelty here in this study are proposing a holistic predictive modeling framework for airport flight delays using AI techniques. The objectives and the contributions of this study are the following.

- To integrate multi-source data including flight schedules, weather conditions, airport-specific features, baggage-specific features and ADS-B signal data.
- To evaluate and benchmark machine learning and deep learning models for delay duration estimation, along with exploring alternative modeling approaches.
- To provide short-term arrival flight delay prediction based on ADS-B data.
- To provide short-term departure flight delay prediction based on flight data and baggage data.
- To introduce novel "Dissimilarity Ratio" feature derived from baggage operations and used as an input during flight delay prediction.
- To utilize explainability methods such as SHAP analysis to identify key delay drivers, offering practical recommendations for airline and airport stakeholders.

In Section 2, we discuss the available solutions, approaches and studies in literature. Then, we introduce the details of our proposed framework along with dataset, AI techniques and use-cases in Section 3. In Section 4, we present our detailed results and discussions. Finally, we conclude our paper in Section 5.

2. RELATED WORK

Accurate prediction of flight delays has become a critical challenge in modern transportation systems, driven by the operational, economic, and environmental implications of disruptions in air traffic. As the aviation industry strives to improve efficiency, minimize costs, and improve passenger satisfaction, data-driven and intelligent forecasting approaches have become increasingly valuable.

Recent advances in machine learning and deep learning have enabled the development of predictive models that can capture complex temporal patterns, non-linear relationships, and contextual factors influencing delays. These models vary in scope and methodology, from large-scale network-wide analyses to airport-specific implementations. They also consider incorporating various data sources, such as historical flight records, weather conditions, and operational parameters. In addition to improving prediction accuracy, recent efforts also emphasize model interpret-ability, feature engineering, and real-world applicability, aiming to create systems that are both technically robust and operationally practical. The following studies illustrate this evolving landscape, showcasing a range of strategies and approaches applied to the flight-delay prediction problem and beyond. Recent studies cluster into three strands: (i) arrival-focused Estimated Time of Arrival (ETA)/delay modeling that leverages ADS-B trajectories and meteorology; (ii) departure-focused prediction of off-block/pushback delays and turnaround dynamics using multi-source operational signals; and (iii) hybrid or network-aware approaches that connect arrival and departure via rotations or data-light pipelines. In parallel, a growing body of work emphasizes xAI to align model outputs with decision-making. Below, we review each strand, synthesize the evidence, explicitly discuss xAI studies, and state the literature gap addressed by the present work

2.1. Arrival-Focused Modeling (ETA / Arrival Delay)

Arrival-focused studies largely exploit ADS-B trajectories, schedules, and weather to estimate short- to mid-horizon ETA or arrival delay. Attention/ Long Short-Term Memory (LSTM)-based models using real-time trajectories with masking for variable-length tracks report high classification accuracy when fusing ADS-B with meteorology and schedule data [17]. Vision-style encodings of trajectories as images reduce terminal-area landing-time MAE by over a third [18]. Classical machine learning baselines remain competitive when enriched with ADS-B and airport/schedule attributes—for instance, Random Forests achieve ≈90% accuracy for binary on-time vs. delayed classification on ADS-B-centric datasets [19], and multi-source Random-Forest regression attains ±5-minute ETA for ≈90% of terminal-area flights [20]. Data-light, online ETA approaches reconstruct partial tracks, match them to historical patterns, and use boosting for speed/path completion at the terminal boundary, supporting real-time deployability [21]. Cross-airport comparisons show deep recurrent models dominating recall in large-scale settings [22]. Network-wide arrival classification with feature engineering and imbalance handling confirms that tree-ensembles and feed-forward nets are strong baselines under rebalancing [12].

2.2. Departure-Focused Modeling (Off-Block, Pushback, Turnaround)

Departure-focused work predicts off-block/pushback deviations—often at 10–120-minute tactical horizons—by combining operational variables (e.g., schedules, resource states), meteorology, and turnaround signals. Explainable multi-source pipelines align prediction with decision-support, with SHAP/Sobol analyses highlighting interpretable drivers (flight type, time-of-day, elevation, weather) and Linear Discriminant Analysis balancing recall and transparency in some airport-level settings [14]. Regional case studies report that boosting methods (CatBoost, XGBoost, LightGBM) deliver strong classification/regression performance; calendar/time features and historical delay rates often outrank weather variables, while weather can provide marginal gains depending on context [15-16]. In our departure context, incorporating baggage-flow temporal signals—particularly a normalized Dissimilarity Ratio that quantifies divergence between observed and baseline offload profiles—improves predictive accuracy and supports actionable thresholds for surge staffing and belt consolidation.

2.3. Hybrid & Network-Aware Approaches (Arrival + Departure, Rotations, Data-Light)

Hybrid and network-aware approaches link arrival and departure by modeling rotations or by learning along the route with minimal covariates. A representative example is the Flight Delay Path Previous-based Machine Learning (FDPP-ML) framework, which restructures schedule records into path sequences and injects ‘flight duration time’ and ‘previous-flight delay’ features—achieving substantial MAE/MSE reductions on 2-hour horizons without hard-to-obtain inputs [13]. Complementary work investigates broader, network-level baselines and US-scale settings [12], motivating joint consideration of arrival-driven constraints and departure-side turnaround risks within a single operational frame.

2.4. xAI for Flight-Delay Prediction

Beyond accuracy, explainability is crucial for operational adoption. Airport-level studies employing xAI report that

global SHAP importance and Sobol sensitivity can surface consistent drivers and support factor prioritization while maintaining competitive predictive performance [14]. Although LIME-style local explanations are conceptually suitable for incident-level review, explicit LIME usage is rarely documented in the aviation delay literature compared to SHAP/Sobol. Overall, xAI has been used to rationalize models in both arrival and departure contexts, yet its integration with formal ablation to quantify the incremental value of specific data sources (e.g., ADS-B vs. meteorology; baggage-flow features vs. generic operational fields) remains uncommon

2.5. Comparative Synthesis

A comparative synthesis regarding related works are summarized in Table 1. It systematically categorizes each referenced work according to several dimensions: category (arrival, departure, hybrid, baseline), prediction horizon (minutes, hours, tactical, etc.), scope (route-level, single airport, network-wide), primary data sources (e.g., ADS-B, weather, schedules, operational signals), methodological approach (e.g., LSTM-Attention, Random Forest, boosting methods), and the practical significance or rationale for each study (why it matters). That overview enables readers to quickly understand the existing approaches in data and methodological diversity perspective, and our respective contributions to the field with this study.

Table 1. Comparative Synthesis of Related Works

Study	Category	Horizon	Scope	Primary data	Method	Why it matters
Chaudhuri et al. [17]	Arrival	Minutes	Route-level	ADS-B + wx + schedule	LSTM-Attention	High accuracy.; trajectories help
Huang et al. [18]	Arrival	Minutes	Single airport	Trajectory images	Deep Learning (vision)	MAE nearly halved
Wells et al. [20]	Arrival	Minutes	Terminal area	Tracks + wx + plan	Random Forest	±5 min for 90%
Hatipoglu and Tosun [16])	Departure	Hours	Single airport	Airport + wx	Boosting methods	Ops features > weather
Alfarhood et al. [15]	Departure	Tactical	National airline	5-yr flight + wx	CatBoost	Best accuracy.; lowest MAE
Pineda-Jaramillo et al. [14]	xAI	Tactical	Single airport	Ops + geo + wx	10 Machine Learning algorithm + SHAP/Sobol	Interpretable drivers
Mamdouh et al. [13]	Hybrid / Data-light	≤2 h	Network-wide	Schedules	FDPP-ML	High gains w/ minimal inputs
Kiliç and Sallan [12]	Baseline	Tactical	US network	Flight + wx	Logistic Regression/ Random Forest/Boosting/Feed-Forward Neural Networks	Strong baselines; rebalancing
Our study	Arrival & Departure	Minutes (arrival), Tactical (departure)	Single Airport	Flight schedules, ADS-B, weather, airport features, baggage features	XGBoost, CatBoost, Random Forest,LightGBM, Gradient Boosting, AdaBoost, Deep Neural Network – (DNN)	Holistic framework integrating multi-source data; introduces novel baggage-derived "Dissimilarity Ratio"; demonstrates improved accuracy and operational relevance for both arrival and departure delay prediction

2.6. Synthesis, Literature Gap, and Contribution

Synthesis. The literature establishes (i) arrival-focused ADS-B–driven ETA/delay prediction with deep sequences and calibrated ensembles [17-22]; (ii) departure-focused modeling where multi-source operational signals and boosting perform well, with interpretable drivers highlighted via xAI [14-16]; and (iii) hybrid/data-light routes that leverage rotations and path features to reduce input demands [12-13].

Gap. Despite progress, three shortcomings persist: (1) lack of a joint, single-airport treatment that simultaneously models arrival (short-horizon, ADS-B-centric) and departure (turnaround-aware) within one pipeline; (2) limited integration of xAI with formal ablation to validate that features ranked as important (e.g., ADS-B approach-phase dynamics, baggage-flow dissimilarity) yield quantified gains when present vs. removed; and (3) scarce scenario-based operationalization that converts predictive gains into actionable playbooks (e.g., dynamic gate

readiness, surge staffing triggers), benchmarked against recent studies.

Contribution. We address this gap by (i) developing a joint framework that covers arrival (ADS-B–driven, short-horizon regression) and departure (multi-source regression including a normalized baggage-flow Dissimilarity Ratio) within the same airport; (ii) presenting source- and feature-group ablation to quantify the incremental value of ADS-B (arrival) and baggage-flow signals (departure); (iii) providing xAI analyses aligned with ablation findings; and (iv) reporting a benchmarking table that situates our results among recent studies while acknowledging dataset/horizon differences.

The main advantage and novelty of this study are based on holistic approach to flight delay prediction, which integrates data from multiple sources and considers both departure and arrival perspectives. While most prior research tends to focus on either departure or arrival delays in isolation, this study addresses both dimensions simultaneously with considering several sources.

3. METHODOLOGY

The primary motivation of this research is to introduce a framework about predictive modeling of airport flight delays from the point of arrival and departure perspectives. In this section, we first outline the general framework of the system. Subsequently, we describe the problem, dataset, feature engineering and AI models used in the experiments per flight direction.

Overall system pipeline is displayed in Figure 1. It consists of two stages as Development and Production within various components. In the Development Stage, data from sources like flights, baggage handling, and ADS-B is collected, pre-processed, and structured into datasets. Key features are extracted and selected for model creation, followed by performance evaluation. In the Production Stage, the trained model is applied to real-time data for inference, and the predictions are visualized through monitoring and reporting tools to support operational decision-making.

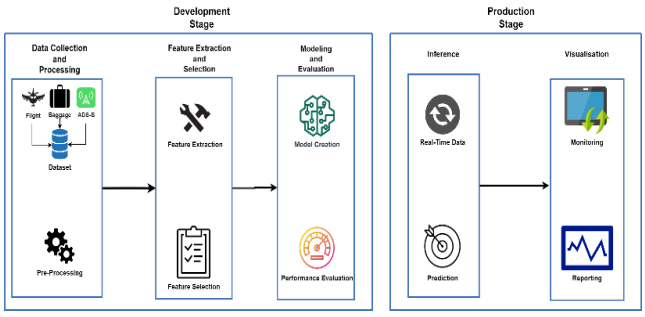

Figure 1. System pipeline

3.1. Arrival Flight Delay Prediction

In this use-case scenario, delay prediction is studied from the point of arrival perspective. In subsections, problem statement, dataset description, feature engineering and model development phases are explained specifically.

3.1.1. Problem Statement

In the aviation sector, the accurate prediction of flight arrival times is vital for enhancing the efficiency of airport operations and passenger satisfaction [24]. While traditional methods predominantly rely on historical data and scheduled flight information, they often prove insufficient in accommodating real-time operational variations [25]. This section introduces a complementary methodology, and an AI model developed to augment longer-term flight delay forecasts. The proposed approach focuses on high-precision, real-time delay prediction by leveraging ADS-B signals from aircraft. It is designed to generate instantaneous predictions that are particularly responsive to dynamic changes, especially during the terminal phases of flight, thereby supporting a range of operational decision-making processes.

3.1.2. Dataset

Airport stakeholders utilize various systems across different operations, including the Flight Management System (FMS) and the Baggage Reconciliation System (BRS). In this part of study, flight-related data (arrivals) were obtained from FMS and external datasets which are related to location of aircraft on route and weather situation of airport.

This study utilized ADS-B signal data from 6,965 flights across 115 routes arriving at İzmir Adnan Menderes Airport, sourced from FlightRadar24 [26], complementing the dataset used in the primary research. Raw ADS-B data inherently provide time-stamped instantaneous flight parameters, including position latitude, longitude, altitude, and speed for each aircraft [27]. Additionally, weather data were derived from Meteorological Aerodrome Report (METAR) reports provided by the meteorological units operating at the airports, allowing us to obtain weather conditions specifically at the time of takeoff and landing. Major features regarding dataset are explained in Table 2.

Table 2. Feature categorization for arrival flight dataset

Category	Feature Name
Core Flight Information	Aircraft Airline Origin Seasonal Flight
Operational Parameters	Route Type Service Type
Time-based Features	Year Month Day Hour

	Minutes Season
Weather Features	Origin Runway Wind Direction Origin Runway Wind Speed Origin Runway Visibility Destination Runway Wind Direction Destination Runway Wind Speed Destination Runway Visibility
ADS-B Features	Latitude Longitude Altitude Gspeed Vspeed Origin Distance Angle

Data quality issues were observed across all systems. In the ADS-B dataset, several inconsistencies were identified, such as missing flight signals over certain time intervals, flights with no recorded data, and unrealistic altitude drops (e.g., sudden 2–3 km losses). These anomalies were filtered out to ensure reliability. For the meteorological data, some time intervals were missing for certain airports; in these cases, temporal interpolation was applied using the closest available reports, and the interpolated weather conditions were aligned with the flight start and end times. Through these preprocessing steps, all datasets were cleaned, synchronized by timestamp, and integrated into a unified structure suitable for downstream analysis and modeling.

3.1.3. Feature Engineering

To enable each signal record to serve as an input instance for the model, an extensive data transformation process was undertaken. This involved calculating the target variable for each ADS-B record: the time remaining (in seconds) from the aircraft's current state to its actual recorded landing time. This procedure yielded a comprehensive dataset comprising approximately 3.266 million samples suitable for model training. Initially, a total of 40 features obtained from multiple data sources were subjected to a comprehensive feature selection process to identify those with a statistically significant impact on the target variable. A diverse set of methods with different underlying principles—namely Correlation Analysis, Chi-Square Test, Information Gain, Recursive Feature Elimination (RFE), CatBoost Feature Importance, and Minimum Redundancy Maximum Relevance (mRMR)—were employed to ensure a balanced evaluation of feature relevance. Subsequently, features identified as important by at least three of these techniques were retained.

Furthermore, the feature engineering phase involved deriving additional variables, such as the angle of the aircraft's current position relative to the departure and arrival airports along its route, the distance to the departure airport, and the distance to the destination airport. This rigorous selection process resulted in 15 key features being carried forward to the modeling phase, enhancing both the robustness and interpretability of the predictive model. For handling categorical variables within the dataset, the CatBoostEncoder was employed to transform these features into a numerical representation.

3.1.4. AI Development

To efficiently process the voluminous dataset, the modeling phase leveraged rapids-cuML, a Graphics Processing Unit (GPU)-accelerated machine learning library [28]. Several ML models (XGBoost, Catboost, Random Forest) are evaluated with using a 5-fold cross-validation strategy to ensure robust and general performance. Subsequent hyperparameter optimization was conducted using the Optuna framework, which employs an efficient search algorithm to explore a wide range of hyperparameter combinations, optimizing key parameters such as the number of estimators, maximum tree depth, and learning rates for each model.

For hyperparameter tuning, as results shown in Table 3, the Optuna framework was employed, utilizing a Bayesian optimization approach to efficiently explore the hyperparameter space. This method was chosen because traditional grid or random search techniques become computationally prohibitive on such large datasets, while Bayesian optimization adaptively focuses on the most promising parameter regions. Optuna was applied consistently across both GPU-accelerated and CPU-based experiments, ensuring optimal configuration and fair performance comparison.

Table 3. Optimized hyperparameters of models

Model	Hyperparameter space
XGBoost	Learning_rate : 0.01 N_estimators : 500 Max depth : 8 Number of leaves : 32 Reg_alpha : 2
CatBoost	Learning_rate : 0.01 N_estimators : 500 Max depth : 8 Number of leaves : 32 Reg_alpha : 2
Random Forest	Learning_rate : 0.01 N_estimators : 500 Max depth : 8

	Number of leaves : 32 Reg_alpha : 2
Random Forest Tuned	Learning_rate : 0.09 N_estimators : 2103 Max depth : 12 Number of leaves : 64 Reg_alpha : 2.44

The computational experiments were performed on a workstation equipped with an NVIDIA GeForce RTX-3060 GPU, featuring 12 GB GDDR6 memory and supporting Compute Unified Device Architecture (CUDA) version 12.2. The system was powered by a 12th Gen Intel® Core™ i9-12900HK processor (14 cores, 20 threads) with a base clock speed of 2.5 GHz (up to 5.0 GHz with Turbo Boost), complemented by 32 GB of DDR4 RAM. This high-performance hardware configuration enabled rapid processing of the dataset and efficient execution of the GPU-accelerated algorithms, balancing computational speed and model accuracy.

The models' performance was assessed using a comprehensive set of complementary metrics to ensure a thorough evaluation. MAE was utilized as the primary metric to quantify the average magnitude of prediction errors, offering a straightforward interpretation of model accuracy in minutes.

3.2. Departure Flight Delay Prediction

In this use-case scenario, delay prediction is studied from the point of departure perspective with considering both flight and baggage data. In subsections, problem statement, dataset description, feature engineering and model development phases are explained specifically.

3.2.1. Problem Statement

Accurate prediction of departure flight delays is essential for efficient surface operations and air traffic flow management. Traditional models, largely based on historical averages and scheduled data, lack responsiveness to real-time disruptions occurring during pre-departure phases. This study proposes a data-driven approach leveraging machine learning techniques and multi-source operational data to generate high-resolution, short-term delay forecasts prior to takeoff, aiming to support proactive decision-making in airport and airline operations.

3.2.2. Dataset

This study examines two distinct datasets containing flight operations and baggage handling information. These datasets were acquired from multiple airports with the help/coordination of local stakeholder of airport operations and presented in this study with all identifying information removed to maintain confidentiality. Due to privacy restrictions, the datasets cannot be publicly distributed. The data collection phase spanned eighteen-month period. Regarding weather data collection process, external Representational State Transfer (REST) Application Programming Interface (API) services are used to populate information based on flight time and location properties.

The flight operations dataset encompasses approximately 144K entries collected over the period. It incorporates comprehensive details including aircraft specifications, temporal data (arrival/departure times), airport identifiers, operational status, and chronological markers. Major features regarding departure flight dataset are explained in Table 4.

Table 4. Feature categorization for departure flight dataset

Category	Feature Name
Core Flight Information	Aircraft Airline Origin Destination Flight Number Seat Capacity Number of Repetitions
Operational Parameters	Route Type Service Type
Time-based Features	Day of Week Hour of Day Date
Weather Features	Origin Temperature Origin Pressure

3.2.3. Feature Engineering

Raw data is transformed into clean data via several pre-processing steps. Missing values are carefully handled through imputation or removal, while outliers are detected using statistical methods such as mean, median and excluded to maintain reliability. Duplicate records are identified based on key identifiers such as flight code, scheduled time, and route, and removed to avoid redundancy. All measurement units are standardized, and time-related features such as scheduled and actual departure or arrival times are converted to a common time zone to ensure temporal alignment across Correlation heat-map is prepared to identify feature dependencies.

In addition to the original features mentioned in Dataset, we applied feature extraction and selection process to transform raw data into meaningful analytical components through several categories.

- Time-based Features: These derived features enhance temporal analysis capabilities.

- Delay Features: A comprehensive set of delay-related features provides multiple perspectives on delay patterns.
- Statistical Features: These features provide statistical insights into delay patterns.
- Baggage Features: The Dissimilarity Ratio [29] provides insights into baggage handling performance, measuring deviations from expected processing times. This is a normalized measure that quantifies how different the baggage accumulation pattern of that specific flight is compared to the typical pattern observed for its corresponding flight code.

The Dissimilarity Ratio for a single flight here is defined as shown in formula 1. Numerator b_{ij} is the absolute difference for flight i at hour j and equals to |Hourly Baggage for Flight i at Hour j−Mean Hourly Baggage for Flight Code at Hour j|. Denominator a_j is the mean hourly baggage for the flight code at hour j. Please note that hour j here refers to delta before flight time.

$$DR_i = \frac{\sum_{j=0}^{T}(b_{ij})}{\sum_{j=0}^{T}(a_j)} \quad (1)$$

Each category of features shown in Table 5 serves specific analytical purposes, enabling both broad operational insights and detailed performance analysis. The combination of these features provides a robust foundation for predictive modeling and operational optimization.

Table 5. Extended Feature Categorization

Category	Feature Name
Time-based Features	Is Weekend Is Peak Hour Season
Delay Metrics	Average Airline Delay Average Aircraft Delay Average Route Delay Average Time Delay Average Day Delay Average Season Delay Route Time Delay Airline Day Delay Average Delay Maximum Delay Median Delay
Baggage Metric	Dissimilarity Ratio

In the context of feature space construction, the final representation comprises 29 distinct components. Categorical variables are transformed using one-hot encoding to preserve nominal relationships without imposing ordinal assumptions, while continuous numerical features are normalized through feature scaling techniques to ensure comparability and enhance model convergence.

3.2.4. AI Development

To develop an effective and reliable framework for predicting flight delays, this study incorporates a range of AI models drawn from two key categories: machine learning (XGBoost, LightGBM, Gradient Boosting AdaBoost, Random Forest) and deep learning (DNN). Each of these approaches offers unique strengths in capturing the complex relationships between input features and delay outcomes, and their complementary use contributes to the robustness and generalization of the proposed predictive system.

The model's robustness was ensured through k-fold cross-validation (k=5) with random splitting. Hyperparameter optimization was conducted using a systematic grid search mechanism with MAE as the scoring metric. The hyperparameter optimization strategy balanced performance requirements with computational efficiency. Rather than exhaustively searching vast parameter spaces, we leveraged domain knowledge and preliminary experiments to focus on promising regions of the hyperparameter space. Optimized hyperparameters regarding AI models are summarized in Table 6.

Table 6. Hyperparameter Optimization

Model	Selected Parameters
XGBoost	N_estimators: 1000 Learning_rate: 0.1 Max_depth: 13
LightGBM	N_estimators: 1500 Learning_rate: 0.1 Max_depth: 20
Random Forest	N_estimators: 500 Min_sample_split: 2 Max_depth: 15
Gradient Boosting	N_estimators: 300 Learning_rate: 0.1 Max_depth: 5
AdaBoost	N_estimators: 100 Learning_rate: 0.1
DNN	Hidden Layers: 3 Neurons: (128,64,32) Activation: ReLU Dropout Rate: 0.2

The computational experiments were conducted on a workstation equipped with a dual-GPU setup comprising two NVIDIA GeForce GTX 1080 Ti units, each offering 11 GB GDDR5X memory and supporting CUDA version 12.2. The system features an Intel Xeon E5-2620 v4 processor (8 cores, 16 threads) operating at a base frequency of 2.10 GHz (up to 3.00 GHz with turbo boost), and is complemented by 62 GB of system RAM. The final model architecture was selected based on both performance metrics and computational efficiency considerations.

The model's performance was evaluated through a comprehensive set of complementary metrics to ensure

thorough assessment. MAE was employed to quantify the average magnitude of prediction errors, providing a direct interpretation of model accuracy. MSE was utilized to assess the variance of prediction errors. The Root Mean Square Error (RMSE) was selected to provide an interpretable metric in the same units as the target variable. The coefficient of determination (R2) was computed to measure the proportion of variance in the dependent variable explained by the model, with values ranging from 0 to 1, where higher values indicate better fit.

4. NUMERICAL RESULTS AND DISCUSSIONS

In this section performance evaluation of numerical results are explained in detail regarding per flight direction view (arrival and departure) similar in Methodology section. Further analysis and discussions are also mentioned.

4.1. Arrival Flight Delay Prediction

To predict flight arrival delays accurately, we evaluated the performance of XGBoost, CatBoost, and Random Forest models using MAE as the primary metric. The models were trained and validated on a comprehensive dataset encompassing flight schedules, ADS-B signals, and time related variables. As presented in Table 7, the Random Forest algorithm outperformed both XGBoost and CatBoost, achieving lower MAE scores on both training and test sets. This superior performance of Random Forest can likely be attributed to its ensemble nature, which leverages multiple decision trees to capture complex, non-linear relationships in the data more effectively than the gradient-boosting frameworks of XGBoost and CatBoost.

Additionally, Random Forest demonstrated better generalization on routes with fewer flights, where data sparsity often challenges predictive models. This robustness is due to its bagging approach, which mitigates overfitting by averaging predictions across diverse trees, enabling stable performance even with limited data. Through rigorous hyperparameter optimization—adjusting key parameters such as the number of estimators, maximum tree depth, and minimum samples per leaf—the tuned Random Forest model achieved an impressive MAE of 2.23 minutes on the test set. This result reflects a significant improvement over the untuned model (MAE of 3.61 minutes) and highlights its superior predictive accuracy.

Table 7. MAE scores on train and test sets

Model	Train Score (MAE)	Test Score (MAE)
XGBoost	6.705160	5.800887
CatBoost	6.028014	5.914732
Random Forest	3.564341	3.612258
Random Forest Tuned	2.412423	2.229232

An analysis of the model's predictive accuracy based on the aircraft's proximity to the destination airport, detailed in Figure 2, revealed that predictions during the descent phase exhibited a comparatively higher error margin. This is attributed to the complex maneuvers aircraft undertake in response to prevailing air traffic conditions near the airport, which can vary significantly and are inherently challenging to predict [30]. The difficulty in precisely capturing these dynamic traffic patterns and the varying queue lengths at different airports consequently influenced model performance. Feature importance analysis of the final model underscored the significance of variables such as the aircraft's distance to the departure and arrival airports, and its angular position relative to them.

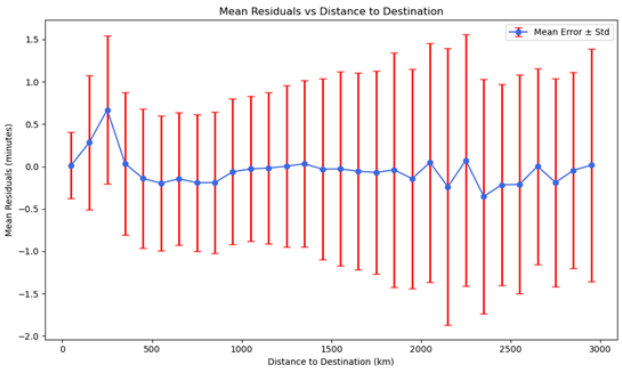

Figure 2. Mean residuals

This ADS-B-based short-term delay prediction framework, by generating near real-time forecasts, complements long-term predictive models beneficial for strategic resource allocation and maintenance scheduling. Its primary contribution lies in facilitating the dynamic optimization of airport operations, such as gate assignments and ground handling coordination, and enhancing air traffic flow management [31].

Although several feature selection techniques were applied to identify the final set of variables, understanding how the best-performing model interprets these variables is equally important. As shown in Figure 3, The SHAP analysis revealed that the most influential feature is dist_to_destination, representing the aircraft’s distance to the destination airport. While this variable may initially appear linearly correlated with the delay outcome, the relationship becomes more complex in practice. During the approach and departure phases, aircraft often perform various maneuvers to align with the correct flight path or adapt to traffic and wind conditions, which weakens the apparent linear dependency and introduces non-linear interactions. Other significant contributors, such as windspeed on destination airports’ runway, altitude, and vertical speed further demonstrate how both environmental factors and dynamic flight characteristics influence delay

predictions. For example, high wind speeds near the runway can extend landing times, while altitude and vertical speed reflect different stages of flight dynamics that the model effectively captures.

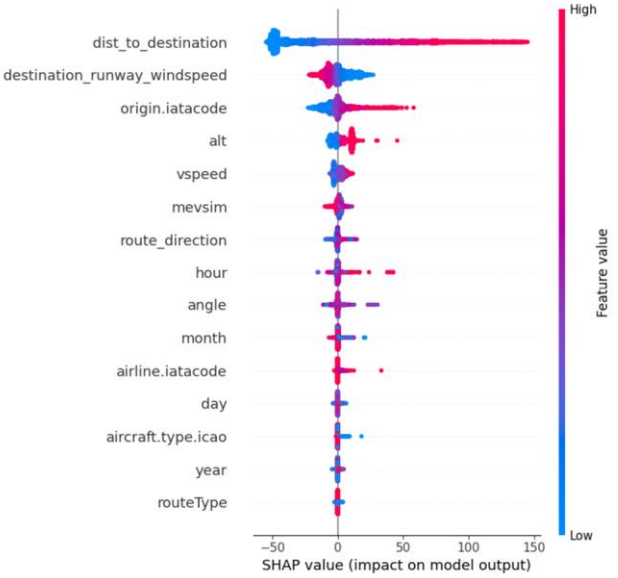

Figure 3. SHAP feature importance analysis

To cross-validate those findings, a Permutation Importance analysis was also conducted. As shown in Figure 4, the resulting feature rankings were largely consistent with the SHAP outcomes, confirming the stability of the model's interpretability results. Both analyses highlight that the model captures complex, non-linear relationships between spatial, temporal, and environmental variables rather than relying on simple additive patterns.

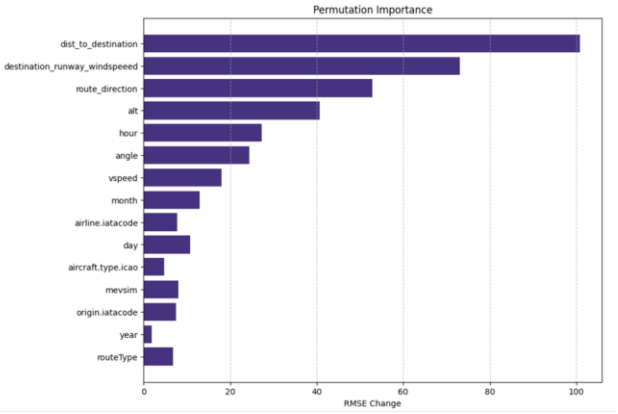

Figure 4. Permutation importance scores

4.2. Departure Flight Delay Prediction

Overall performance comparison is shown in Table 8. We would like to separate results with and without Dissimilarity Ratio where it is a cross-feature providing insight from baggage dataset. XGBoost emerged as the superior performer among all tested models, demonstrating exceptional predictive capabilities with the highest absolute performance metrics especially via minimum Test MAE score as 7.6 minutes. LightGBM demonstrated the most balanced and robust performance profile among all tested models. The relatively poor performance of the DNN (R^2 Score: 0.2742) can be attributed to several factors such as feature properties and representations. The conventional tree-based models, including Random Forest and Gradient Boosting, demonstrated consistent but moderate performance improvements with the integration of the Dissimilarity Ratio, achieving R^2 scores of 0.4461 and 0.4434 respectively.

We would like to focus on evaluating the impact of the newly introduced Dissimilarity Ratio feature, which was extracted from the baggage dataset to capture cross-relational patterns among variables. The results indicate that the inclusion of this feature contributes positively to model performance across almost all evaluated models and metrics. Notably, the most significant improvement was observed in the XGBoost model, where the MSE decreased by up to 24.97%.

Table 8. Performance comparison of models

Model	Metric	With Dissimilarity	Without Dissimilarity	Improvement
XGBoost	Train MAE	**4.2320**	4.8787	+13.26%
	Test MAE	**7.6220**	7.8828	+3.31%
	Train MSE	33.1772	44.2155	**+24.97%**
	Test MSE	112.1498	119.7797	+6.37%
	Train RMSE	5.7600	6.6495	+13.38%
	Test RMSE	10.5901	10.9444	+3.24%
	Train R^2	0.8635	0.8251	+4.65%
	Test R^2	0.5375	0.5258	+2.22%
LightGBM	Train MAE	6.4034	6.8377	+6.35%
	Test MAE	7.8514	8.1218	+3.33%
	Train MSE	76.2445	87.2403	+12.60%
	Test MSE	118.188	126.6679	+6.69%
	Train RMSE	8.7318	9.3403	+6.51%
	Test RMSE	10.8714	11.2547	+3.41%
	Train R^2	0.6862	0.6550	+4.76%
	Test R^2	0.5126	0.4985	+2.83%
	Train MAE	8.0472	8.3596	+3.74%
	Test MAE	8.4194	8.6475	+2.64%

Random Forest	Train MSE	121.186	131.2059	+7.64%
	Test MSE	134.2991	142.223	+5.57%
	Train RMSE	11.0085	11.4545	+3.89%
	Test RMSE	11.5887	11.9257	+2.83%
	Train R^2	0.5013	0.4811	+4.20%
	Test R^2	0.4461	0.4370	+2.08%
Gradient Boosting	Train MAE	8.3291	8.5411	+2.48%
	Test MAE	8.4234	8.6562	+2.69%
	Train MSE	132.308	138.946	+4.78%
	Test MSE	134.9552	143.0988	+5.69%
	Train RMSE	11.5025	11.7875	+2.42%
	Test RMSE	11.617	11.9624	+2.89%
	Train R^2	0.4555	0.4505	+1.11%
	Test R^2	0.4434	0.4335	+2.28%
AdaBoost	Train MAE	10.1921	10.4827	+2.77%
	Test MAE	10.1882	10.4230	+2.25%
	Train MSE	167.3839	175.697	+4.73%
	Test MSE	166.121	174.3044	+4.70%
	Train RMSE	12.9377	13.2551	+2.39%
	Test RMSE	12.8888	13.2024	+2.38%
	Train R^2	0.3112	0.3051	+2.00%
	Test R^2	0.3149	0.3100	+1.58%
DNN	Train MAE	9.4300	9.6000	+1.77%
	Test MAE	10.3100	10.5000	+1.81%
	Train MSE	223.1600	234.0900	+4.67%
	Test MSE	258.1100	263.2000	+1.93%
	Train RMSE	14.9400	15.3000	+2.35%
	Test RMSE	16.0700	16.2200	+0.92%
	Train R^2	0.3590	0.3658	-1.86%
	Test R^2	0.2742	0.2718	+0.88%

Generally, a flight is considered delayed if it arrives/departs 15 minutes or more after its scheduled time [32]. Many aviation studies adopt this threshold to distinguish between on-time and delayed flights for consistency and comparability. Regarding this, as a second phase we would like to study to compare real delay values with predicted ones applied best-performing XGBoost model.

For the training set shown in Figure 5, the model demonstrates strong discriminative power with 86% accuracy, showing balanced performance across both delay categories. The model maintains similar precision and recall values for both delay predictions (0.84-0.86) and no-delay predictions (0.86-0.87).

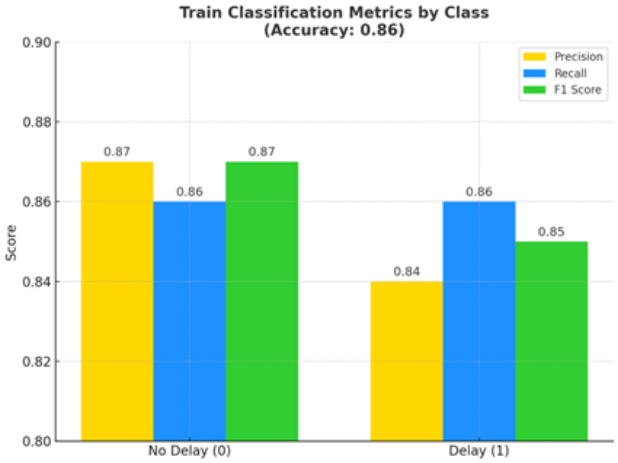

Figure 5. Train set classification

When evaluated on the test set shown in Figure 6, the model achieves 76% accuracy, with a noticeable but reasonable decrease in performance. The no-delay predictions maintain higher precision (0.80) but lower recall (0.75), while delay predictions show lower precision (0.73) but higher recall (0.79). The dataset is relatively balanced, with a slight majority of no-delay cases in both training (62,139 vs 54,356) and test sets (15,450 vs 13,674). This binary transformation of our regression results provides valuable insights into the model's practical utility in predicting significant delays, defined as those exceeding 15 minutes.

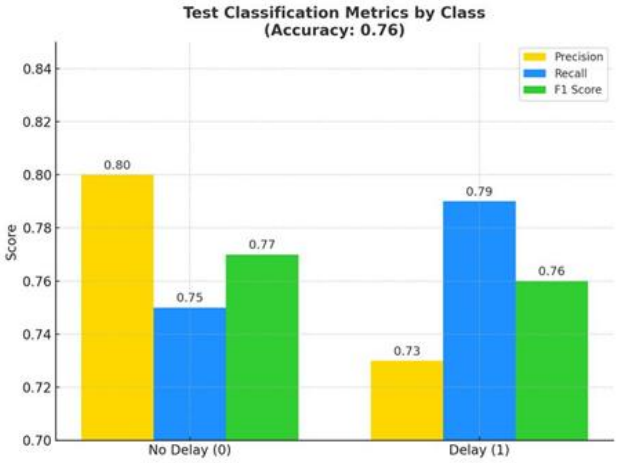

Figure 6. Train set classification

To enhance the interpretability of our predictive models and identify key delay drivers, we employed SHAP analysis (for top 20 features) on the best-performing XGBoost model for departure delay prediction as shown in Figure 7. SHAP values provide a unified framework for

understanding feature contributions to individual predictions and overall model behavior.

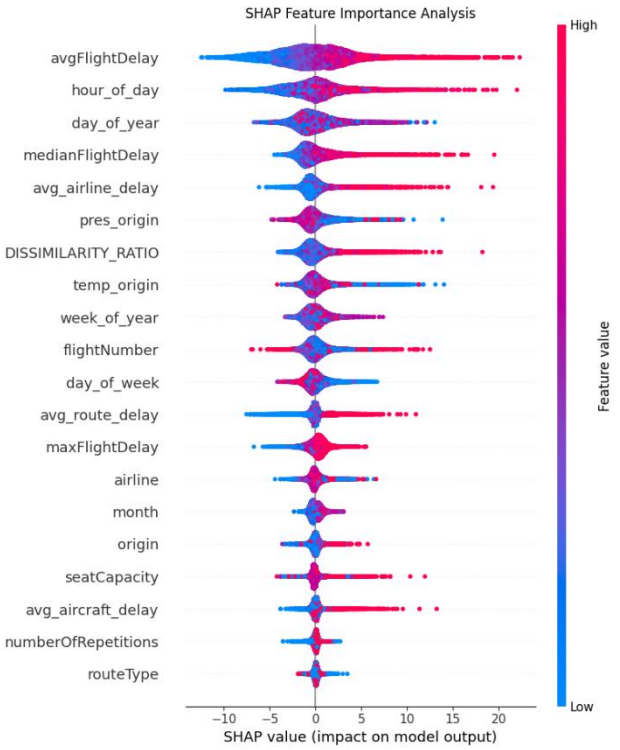

Figure 7. SHAP feature importance analysis

The SHAP analysis reveals several critical insights into the factors driving departure flight delays. Some of the major features are described within the following bullet points.

- avgFlightDelay: Historical average flight delays emerge as the strongest predictor. This indicates that flights with historically poor punctuality performance are significantly more likely to experience delays.
- hour_of_day: Time of day shows substantial impact, with peak hours (morning and evening rush periods) contributing positively to delay predictions, while off-peak hours show negative contributions.
- day_of_year: Seasonal patterns demonstrate clear influence, with certain periods (likely holiday seasons and summer months) showing higher delay propensity.
- DISSIMILARITY_RATIO: Novel introduced feature ranks in 7th place, demonstrating significant predictive power. Higher dissimilarity ratios (indicating deviation from normal baggage processing patterns) consistently contribute to increased delay predictions.
- medianFlightDelay and avg_airline_delay: Airline-specific performance metrics show strong predictive capability, highlighting the importance of carrier operational efficiency.
- pres_origin and temp_origin: Weather conditions at origin airports contribute moderately to delay predictions, with extreme values in either direction increasing delay likelihood.

4.3. Discussions

A benchmarking study is prepared as shown in Table 9. Our study demonstrates competitive performance compared to existing literature, with the Random Forest model achieving an MAE of 2.23 minutes for arrival prediction—comparable to Reddy et al. [8] (2.06 minutes, XGBoost) and Singh et al. [11] (2.2 minutes, Linear Regression). For departure prediction, the XGBoost model obtained an MAE of 7.62 minutes.

Table 9. Benchmarking of Studies

Study	Model	MAE (minutes)
Our Study (Arrival)	Random Forest	2.23
Our Study (Departure)	XGBoost	7.62
Reddy et al. [8] (Arrival)	XGBoost	2.06
Singh et al. [11] (Arrival)	Linear Regression	2.2
Alfarhood et al. [15] (Arrival)	CatBoost	12.19

The operational contribution of our integrated flight delay prediction framework lies in its ability to transform data-driven insights into actionable operational decisions. When an airport faces challenging weather conditions, the Random Forest model using ADS-B signals provides early warning of compressed arrival patterns. This advance notice allows operations teams to mobilize ground resources, adjust staffing, and reconfigure gates before aircraft land. For departure operations, the XGBoost model incorporating baggage handling data identifies potential delays well before scheduled departure times. This early intelligence enables airlines to implement targeted interventions: adjusting ground handling priorities, addressing potential crew duty limitations, and making informed decisions about connecting flights.

5. CONCLUSION

The primary aim of this study was to develop a comprehensive framework to predict flight delays at airports using AI techniques. Through our analysis, which incorporated a mix of ADS-B data and multi-source information on flight operations, weather, and baggage handling, we have identified key patterns that significantly impact delay times.

For arrival flight delay prediction, our research evaluated multiple machine learning models and established that the Random Forest model significantly outperforms other approaches, including XGBoost and CatBoost, particularly for short-term arrival delay predictions. We quantitatively demonstrated improved accuracy with the tuned Random Forest model achieving test MAE score as of 2.23 minutes compared to standard models, representing a substantial improvement in prediction capability during critical flight phases such as descent. In parallel, we developed departure delay prediction framework where XGBoost emerged as the optimal model, showing substantial performance improvements over alternative approaches. We quantified these performance gains across multiple metrics including R^2, MAE, and RMSE, providing clear evidence of our model's superiority for departure delay forecasting. This dual-focused approach to both arrival and departure delays represents a more holistic treatment of the flight delay prediction challenge than is typically found in the literature. As a next step, evaluating isolated impact of ADS-B and baggage data can be promising as an ablation study.

One of our most innovative contributions was the introduction of the "Dissimilarity Ratio" feature derived from baggage handling data. We demonstrated consistent improvement across all models when incorporating this feature, with up to 24.97% improvement in train MSE score. These findings contribute to the field by confirming the utility of machine learning in aviation management, aligning with current technological advancements, and refining existing delay prediction models through the incorporation of novel variables, such as the "Dissimilarity Ratio" in baggage handling.

This study demonstrates significant potential for generalization across diverse airport environments and aviation networks. While our implementation focused on specific airport data, the underlying methodology and architecture have been designed with broader applicability in mind. The feature engineering process, especially the development of the "Dissimilarity Ratio" regarding baggage handling, provides a blueprint that can be replicated at facilities with comparable baggage systems. For network-level implementation, our dual focus on both arrival and departure delay prediction creates natural integration points within larger aviation systems. To facilitate practical generalization, several adaptation strategies would be beneficial such as fine-tuning based on airport-specific data, guidance identified by the feature importance rankings.

This research bears significant theoretical and practical implications. It supports the further development and integration of AI-driven models in airport operations, potentially guiding policy enhancements that improve efficiency and passenger satisfaction. Recognizing limitations such as the variability of terminal operations and external factors not covered by our model, our study remains a valuable reference despite potential constraints in fully generalizing the results.

Future research should focus on expanding and augmenting the dataset to include a broader range of external variables, improving model robustness. Emerging AI technologies such as Generative AI (GenAI) and Large Language Model (LLM) can be integrated as an interface for talking with flight records and commenting/assisting on results. Additionally, adaptive learning represents a promising future direction for allowing the flight delay prediction models continuously evolve and improve in response to changing conditions and new data patterns. Such efforts will offer a more comprehensive understanding of delay mechanisms and bolster proactive operational strategies in aviation. Finally, the implementation of predictive modeling methodologies for baggage volume forecasting presents a significant opportunity for aviation stakeholders to optimize operational efficiency and mitigate flight delays This operation actions can be deploying dynamic staff allocation tools by airport managers that adjust baggage handler assignments based on predicted volumes with granular horizons, enabling proactive rerouting by airline staff regarding baggage system capacity alerts.

REFERENCES

[1] F. Erdem, T. Bilgiç, “Airline delay propagation: Estimation and modeling in daily operations”, *Journal of Air Transport Management*, 115, 102548, 2024.

[2] W. R. Patterson, “Federal Aviation Administration (FAA)”, **The Handbook of Homeland Security**, CRC Press, Boca Raton, 65–70, 2023.

[3] EUROCONTROL Central Office of Delay Analysis, **CODA Digest: All-Causes Delays to Air Transport in Europe Annual** 2022, EUROCONTROL, Brussels, 2023.

[4] G. Enea, T. Reynolds, M. Weber, R. D. Codina, D. Schaefer, “Analysis of weather-driven air traffic management challenges for major US and European airports”, **SESAR Innovation Days**, 2024.

[5] L. Carvalho, A. Sternberg, L. Maia Goncalves, A. Beatriz Cruz, J. A. Soares, D. Brandão, E. Ogasawara, “On the relevance of data science for flight delay research: a systematic review”, Transport Reviews, 41(4), 499–528, 2021.

[6] Q. Li, R. Jing, “Flight delay prediction from spatial and temporal perspective”, Expert Systems with Applications, 205, 117662, 2022.

[7] M. Zoutendijk, M. Mitici, “Probabilistic flight delay predictions using machine learning and applications to the flight-to-gate assignment problem”, Aerospace, 8(6), 152, 2021.

[8] R. T. Reddy, P. B. Pati, K. Deepa, S. T. Sangeetha, “Flight delay prediction using machine learning”, **2023 IEEE 8th International Conference for Convergence in Technology (I2CT)**, 1–5, 2023.

[9] W. Jmoona, M. U. Ahmed, M. R. Islam, S. Barua, S. Begum, “Explaining the unexplainable: Role of XAI for flight take-off time delay prediction”, **IFIP International Conference on Artificial Intelligence Applications and Innovations**, 81–93, 2023.

[10] A. Degas, M. R. Islam, C. Hurter, S. Barua, H. Rahman, M. Poudel, D. Ruscio, M. U. Ahmed, S. Begum, M. A. Rahman, "A survey on artificial intelligence (AI) and explainable AI in air traffic management: Current trends and development with future research trajectory", Applied Sciences, 12(3), 1295, 2022.

[11] J. Singh, M. D. Jayaprakash, R. Agarwal, "Flight delay prediction for Indian air carriers with explainable artificial intelligence", **2022 Third International Conference on Smart Technologies in Computing**, Electrical and Electronics (ICSTCEE), 1–6, 2022.

[12] K. Kılıç, J. M. Sallan, "Study of delay prediction in the US airport network", Aerospace, 10(4), 342, 2023.

[13] M. Mamdouh, M. Ezzat, H. A. Hefny, "A novel intelligent approach for flight delay prediction", *Journal of Big Data*, 10(1), 179, 2023.

[14] J. Pineda-Jaramillo, C. Munoz, R. Mesa-Arango, C. Gonzalez-Calderon, A. Lange, "Integrating multiple data sources for improved flight delay prediction using explainable machine learning", Research in Transportation Business & Management, 56, 101161, 2024.

[15] M. Alfarhood, R. Alotaibi, B. Abdulrahim, A. Einieh, M. Almousa, "Predicting flight delays with machine learning: A case study from Saudi Arabian Airlines", *International Journal of Aerospace Engineering*, 2024(1), 3385463, 2024.

[16] İ. Hatipoğlu, Ö. Tosun, "Predictive modeling of flight delays at an airport using machine learning methods", Applied Sciences, 14(13), 5472, 2024.

[17] T. Chaudhuri, S. Zhang, Y. Zhang, "Attention-based deep learning model for flight delay prediction using real-time trajectory", Proceedings of the SESAR Innovation Days (SIDs) 2024, Paper 006, 2024.

[18] L. Huang, S. Zhang, Y. Zhang, Y. Zhang, Y. Yin, "Aircraft landing time prediction with deep learning on trajectory images", arXiv preprint arXiv:2401.01083, 2024.

[19] G. Swetha, S. Sharmila, M. Parkavi, S. Preetha, S. Gayathri, "Flight delay prediction using ML", *International Journal of Advanced Research in Science, Communication and Technology*, 4(1), 183–192, 2024.

[20] J. Z. Wells, T. G. Puranik, K. M. Kalyanam, M. Kumar, "Prediction of aircraft estimated time of arrival using a supervised learning approach", IFAC-PapersOnLine, 56(3), 43–48, 2023.

[21] D. Ni, Y. Gao, S. Yin, "A data-light and trajectory-based machine learning approach for the online prediction of flight time of arrival", Aerospace, 10(7), 630, 2023.

[22] A. Ayaydın, M. A. Akcayol, "Deep learning based forecasting of delay on flights", *Bilişim Teknolojileri Dergisi*, 15(3), 239–249, 2022.

[23] B. Arslan, H. Tiryaki, "Prediction of railway switch point failures by artificial intelligence methods", *Turkish Journal of Electrical Engineering and Computer Sciences*, 28(2), 1044–1058, 2020.

[24] J. Smith, A. Doe, "The impact of flight prediction accuracy on airport efficiency and passenger satisfaction", *Journal of Air Transport Management*, 88, 101867, 2020.

[25] B. Johnson, C. Williams, "A review of flight delay prediction methods: Challenges and opportunities", Transportation Research Part C: Emerging Technologies, 130, 103332, 2021.

[26] P. Davies, L. White, "Leveraging public ADS-B data for air traffic research and analysis", **Proceedings of the IEEE International Conference on Aerospace,** 2019.

[27] International Civil Aviation Organization (ICAO), Annex 10: Aeronautical Telecommunications, Volume IV: Surveillance and Collision Avoidance Systems, ICAO Standards and Recommended Practices, 2018.

[28] S. M. Ghazimirsaeed, Q. Anthony, A. Shafi, H. Subramoni, D. K. Panda, "Accelerating GPU-based machine learning in Python using MPI library: A case study with MVAPICH2-GDR", 2020 IEEE/ACM Workshop on Machine Learning in High Performance Computing Environments (MLHPC) and Workshop on Artificial Intelligence and Machine Learning for Scientific Applications (AI4S), November 2020, 1–12, 2020.

[29] N. Gozuacik, A. Tekinbas, E. Sag, O. Adiguzel, S. Malkos, "Temporal Pattern Analysis of Baggage Impact on Flight Operations," in Proc. 14th Int. Conf. Pattern Recognition Applications and Methods (ICPRAM), SciTePress, 2025, pp. 831–838, doi: 10.5220/0013372800003905.

[30] D. Lee, E. Kim, "Modeling and prediction of air traffic congestion in terminal maneuvering areas", *Journal of Advanced Transportation*, 2022, 1–15, 2022.

[31] R. Brown, S. Green, "Benefits of enhanced predictability for airport resource management and air traffic flow management", Aerospace Science and Technology, 103, 105943, 2020.

[32] T. K. Huynh, T. Cheung, C. Chua, "A systematic review of flight delay forecasting models", **2024 7th International Conference on Green Technology and Sustainable Development (GTSD)**, July 2024, 533–540, 2024.