

Prediction of Arrival Runway Occupancy Time and Exit Taxiway Using ADS-B Trajectories

Kevin Hänggi ^{*,1} Jeremy Wilde,² and Manuel Waltert ¹

¹Centre for Aviation, Zurich University of Applied Sciences, Winterthur, Switzerland

²Zurich University of Applied Sciences, Winterthur, Switzerland

*Corresponding author: kevin.haenggi@zhaw.ch

(Received: 31 Oct 2025; Revised: 30 Mar 2026; Accepted: 21 May 2026; Published: 4 Jun 2026)

(Editor: Ramon Dalmau; Reviewers: Gabriel Jarry and Yutong Chen)

Abstract

The expected future growth in aircraft movements will require airports to increase runway capacity, which is often constrained, among other factors, by aircraft arrival runway occupancy time (AROT) and by the rapid exit taxiway (RET) selected by pilots. Existing prediction approaches for AROT and RET selection rely mostly on proprietary Advanced Surface Movement Guidance and Control System (A-SMGCS) or radar data and often ignore temporal context in trajectory patterns, leaving gaps for operationally relevant applications. In this study, we used Automatic Dependent Surveillance–Broadcast (ADS-B) trajectory data sourced from the OpenSky Network for flights arriving at Zurich Airport in the years 2024–2025 to train two machine learning models: a LightGBM model for RET prediction and a neural network combining time-invariant features with time-variant ADS-B trajectory snippets for AROT prediction. The results of both models were within the range reported in the literature. The RET prediction model achieved a weighted accuracy of 79.4 %, while the AROT prediction model yielded a mean absolute error of 3.95 s and a root mean square error of 5.01 s. These findings demonstrate that ADS-B-based models can support air traffic controllers in reducing separation between arriving aircraft, thereby potentially enhancing runway arrival throughput at aerodromes.

Keywords: Arrival Runway Occupancy Time; Rapid Exit Taxiway; Automatic Dependent Surveillance-Broadcast; Machine Learning; Trajectory Snippets

Abbreviations: ADS-B: Automatic Dependent Surveillance–Broadcast AROT: Arrival Runway Occupancy Time ATC: Air Traffic Controller RET: Rapid Exit Taxiway ROT: Runway Occupancy Time

1. Introduction

The number of aircraft movements worldwide continues to grow significantly [1]. This steady increase in air traffic exerts growing pressure on airport infrastructure, where the potential for expanding runway systems is often limited due to spatial constraints and high investment costs. As a result, airport capacity is frequently restricted, and physical expansion may not always be economically viable [2]. To address this challenge, stakeholders in air traffic management and airport operations often identify and implement measures that optimise the utilisation of existing runway systems. One approach to increasing runway capacity, which is also cost-effective, is to address the factors that limit maximum arrival throughput. This term refers to the number of arriving aircraft movements that can be handled on a runway system within one hour. The main limiting factors include the minimum radar separation and the minimum wake turbulence separation between approaching pairs

of aircraft, as well as the runway occupancy time (ROT) of each aircraft. The applicable separation distance between consecutive arrivals, and thus the achievable runway throughput, is determined by the most restrictive of these three factors. When MRS and WT are reduced through operational or technical improvements, ROT frequently becomes the dominant factor limiting runway capacity [3].

The arrival runway occupancy time (AROT) is defined as the time interval between the moment an aircraft crosses the runway threshold and the moment it completely vacates the runway [3]. To maximise runway throughput, aircraft would ideally be sequenced such that the time between two successive runway occupancies equals zero, meaning the next aircraft crosses the threshold precisely when the preceding one vacates the runway. In practice, however, this is not feasible due to safety requirements and uncertainty related to both the AROT and the rapid exit taxiway (RET) selection of the arriving aircraft. Therefore, air traffic controllers (ATC) maintain time buffers between consecutive landings to ensure that the preceding aircraft has vacated the runway before the next aircraft touches down [4]. The magnitude of these buffers depends on several factors, including prevailing weather conditions, controller experience, aircraft type, and whether the arriving flight is operated by a home-based airline. In addition, pilot decisions and airline procedures can influence both AROT and RET; for instance, a pilot may remain longer on the runway to use a preferred exit and thereby reduce taxi distance. Runway characteristics such as length, slope, and RET geometry further affect AROT [4]. As a result of these interacting factors, ATC must apply conservative buffer times to mitigate risks such as missed approaches or runway incursions that could arise from inaccurate AROT or RET predictions [2]. Figure 1 shows the time intervals between a leading aircraft vacating runway 14 and the following aircraft crossing the threshold upon landing at Zurich Airport. The majority of intervals are concentrated between 20 and 60 s, indicating potential for optimisation in runway utilisation.

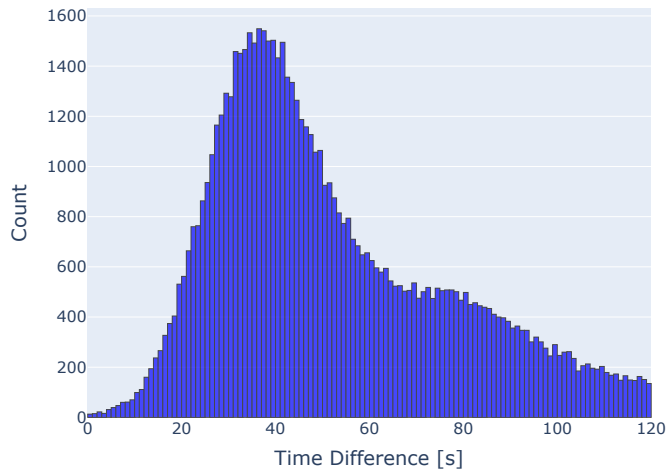


Figure 1. Histogram of Time Between Runway Vacation and Next Threshold Crossing on Runway 14 at Zurich Airport (1 March 2024 to 31 March 2025)

Previous research has investigated the prediction of both AROT and RET selection using a variety of data-driven approaches. Herrema et al. [2] applied a Gradient Boosting algorithm to distinguish between procedural and non-procedural runway exits at Vienna Airport. The data used in this study included radar, Advanced Surface Movement Guidance and Control System (A-SMGCS), wind profiler, Snow Notice to Airmen (SNOWTAM), Sonic Detection and Ranging (SODAR), and Meteorological Aerodrome Report (METAR) information. The model achieved an accuracy of 79 %. For

Singapore Changi Airport, Woo et al. [5] employed an Extreme Gradient Boosting (XGBoost) model supported by Local Interpretable Model-agnostic Explanations (LIME) to predict the RET used by arriving aircraft. The model was trained on A-SMGCS, operational, and METAR data collected over a two-month period and achieved an accuracy of 80.76 %, a precision of 86.99 %, a recall of 80.76 %, and an F1-score of 82.91 %. Martínez et al. [6] developed a resilient neural-network-based prediction framework for AROT using historical operational data from Barcelona International Airport. The authors introduced a multi-stage prediction framework with four temporal update stages, corresponding to flight planning, stand assignment, the latest METAR update, and the final approach, achieving an absolute accuracy of 76.31 % within ± 7 s and a percentage accuracy of 87 % within ± 20 % of the real AROT. Jun, Wei, and Alam [7] applied a decision-tree-based regressor for AROT prediction at Changi Airport using A-SMGCS data with a reference point 4.4 NM before the runway threshold. The model achieved a test root mean square error (RMSE) of 5.96 s. Stempfeler et al. [8] applied an XGBoost regressor to predict AROT at Zurich Airport. Separate models were trained for four prediction horizons, ten minutes before landing, 8 NM, 2 NM, and at the runway threshold, using three years of radar, A-SMGCS, and METAR data. The best model, trained on data available at the threshold, achieved a coefficient of determination (R^2) of 0.582. Nguyen et al. [9] proposed a generalised AROT prediction model trained on data from Ronald Reagan Washington National, Miami International, and Phoenix Sky Harbour Airports. The model achieved mean absolute errors (MAE) between 3.7 s and 6.8 s across the three airports and reduced AROT uncertainty by 32 % to 47 %. Mirmohammadsadeghi and Trani [10] applied a neural-network algorithm based on two years of Airport Surface Detection Equipment Model X (ASDE-X) data from 35 airports across the United States to predict AROT for arriving flights, achieving a weighted average R^2 of 0.90 across 14 aircraft types and eight airports.¹

The existing literature on predicting AROT and RET still leaves considerable room for improvement. Most studies rely on A-SMGCS or radar track data, which are proprietary and therefore not publicly accessible to researchers. In contrast, Automatic Dependent Surveillance–Broadcast (ADS-B) data are openly available through platforms such as the OpenSky Network [11]. Since 2020, civil aviation aircraft in both Europe and the United States have been required to be equipped with ADS-B transmitters [12]. This widespread implementation makes ADS-B a suitable and scalable data source for predicting both AROT and RET selection. In addition, most existing studies base AROT prediction on a single trajectory point recorded at a specific moment in time. To the best of our knowledge, trajectory snippets representing the aircraft's motion in a certain interval preceding that point have not yet been investigated in this context. These snippets capture recent changes in altitude and speed and can improve prediction accuracy by providing temporal context and enhancing robustness to outliers. Furthermore, only a few studies explicitly address RET prediction. Those that do typically focus on distinguishing procedural from non-procedural runway exit usage rather than predicting the specific RET selected by the pilots.

In light of these gaps, our study investigates two main aspects. First, we examine the potential of ADS-B trajectory data for developing a predictive model of RET selection. Second, we explore the use of ADS-B trajectory snippets for predicting AROT, thereby extending the existing body of knowledge. More broadly, the aim of this paper is to apply state-of-the-art machine learning methods to a practically relevant operational problem. By addressing these two aspects, the study aims to contribute to more reliable arrival management practices. In this context, our approach provides direct operational benefits by reducing management uncertainty for air traffic controllers, which enables smaller separation buffers between arrivals and ultimately increases runway arrival throughput.

¹Airport Surface Detection Equipment – Model X (ASDE-X) is a surveillance system combining radar, multilateration, and satellite technology to monitor aircraft and vehicle movements on the ground. Its data sources include surface surveillance radar, multilateration sensors, airport surveillance radars such as Mode S, ADS-B sensors, and flight plan data from the terminal automation system.

The remainder of this paper is structured as follows. In Section 2, we describe the data, feature engineering, and modelling approach used to predict RET selection and AROT based on ADS-B trajectories obtained from the OpenSky Network. In Section 3, we present the model results, while Section 4 discusses the findings, their operational implications, and the limitations of our approach. Finally, Section 5 summarises the main conclusions and provides an outlook for future research.

2. Methods

This section presents our methodology for utilising ADS-B data sourced from the OpenSky Network to predict AROT and RET on runway 14 at Zurich Airport. Our method consists of three steps. First, Section 2.1 describes the data collection and pre-processing steps. Next, we split the dataset into two equally sized parts: The first half, referred to as the RET dataset, is used in Section 2.2 to train and validate the RET prediction model. The second half, referred to as the AROT dataset, serves as a test set for the RET model. The resulting prediction probabilities are then added to the AROT dataset, which we subsequently use in Section 2.3 to build, validate, and test the AROT prediction model.

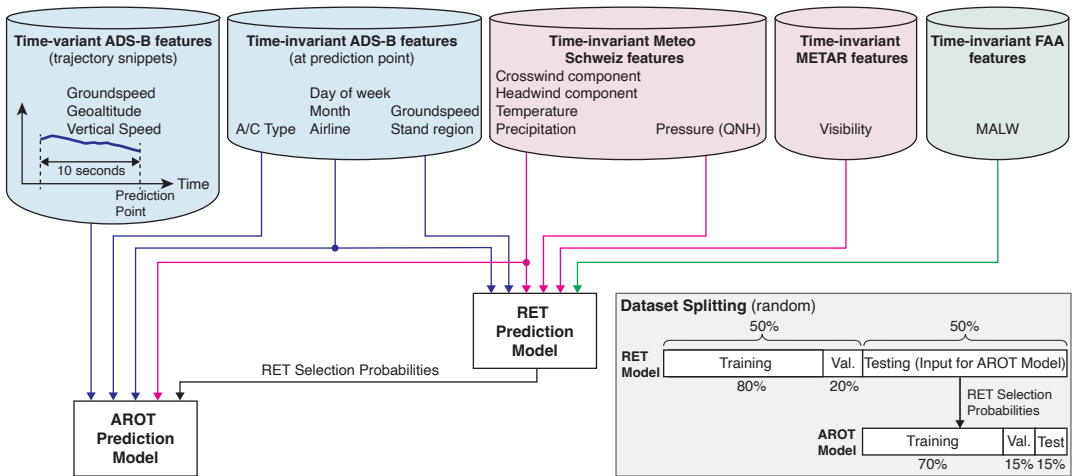


Figure 2. Overview of Feature Inputs and Dataset Splitting for the RET and AROT Prediction Models

2.1 Data Collection and Pre-Processing

2.1.1 Selection of Airport

To develop prediction models for AROT and RET selection, an airport with sufficient surface and near-surface ADS-B coverage was required. In February 2024, an additional ADS-B receiver contributing to the OpenSky Network was installed at Zurich Airport, ensuring high-quality coverage of ground and near-ground traffic movements. For this reason, we selected Zurich Airport as the study location. Moreover, we decided to restrict our analysis to arrivals on runway 14 in order to avoid potential influences arising from differing runway characteristics.

We obtained ADS-B trajectory data from the OpenSky Network for the period from 1 March 2024 to 31 March 2025 using the `history()` function provided by the `traffic` library [13]. To ensure that the dataset contained only arrivals on runway 14 at Zurich Airport, we defined a polygon around the airport and extended it northwards to capture aircraft approximately ten minutes before landing. Duplicate entries were removed, and data points above 8000 ft were excluded, as they lay outside the altitude range relevant for determining AROT and RET. We then resampled the remaining data to a frequency of 1 Hz to ensure consistent temporal resolution and to prevent inaccuracies in AROT

and RET determination due to missing values. Finally, we excluded all flights that were not aligned with runway 14 for at least 30 s, thereby ensuring that the final dataset only included actual landings on runway 14.

2.1.2 Determination of Arrival Runway Occupancy Time

To accurately measure and predict AROT, it is essential to identify precisely when an aircraft is on the runway and when it has vacated it. We treated each aircraft as a single point located at the position reported by its Global Navigation Satellite System (GNSS) receiver. This approach was necessary because the GNSS antenna position varies between aircraft types, and, to our knowledge, no open access database exists that specifies its exact location. As a result, the precise positions of the nose and tail cannot be derived directly from ADS-B messages. To determine AROT based on ADS-B data, we defined three polygons: a *runway polygon*, a *runway-vacated polygon*, and a *go-around polygon*. The *runway polygon*, which closely reflects the geometric dimensions of the actual runway, defines the area within which an ADS-B data point is considered relevant for the AROT calculation. The *runway-vacated polygon* is used to determine when an aircraft exits the runway after landing, while the *go-around polygon* assists in detecting flights that initiated a go-around. The locations of these polygons for runway 14 at Zurich Airport are shown in Figure 3. The runway polygon extends from the threshold to the end of runway 14, with a width corresponding to that of the runway. We determined the coordinates of all polygons using *Google Maps*. The approach could in principle be generalised by automatically generating polygons from digital Aeronautical Information Publication data or OpenStreetMap features to facilitate application across multiple airports. However, the feasibility of such automation depends on the availability and quality of the underlying data. The defined polygons were then used to remove all aircraft trajectory data points lying outside their boundaries from the dataset.

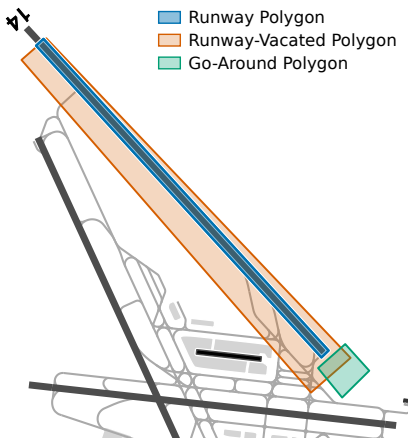


Figure 3. Visualization of Polygons for Runway 14 at Zurich Airport

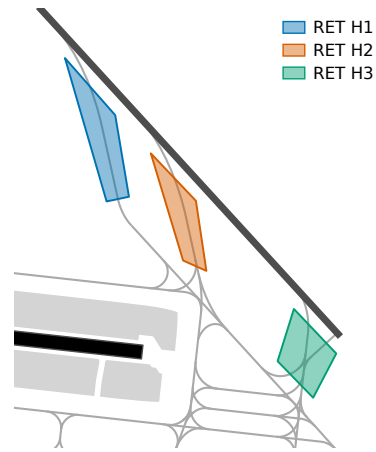


Figure 4. Visualization of RET Polygons for Runway 14 at Zurich Airport

To determine the AROT of each flight, we applied a multi-step procedure. First, we calculated a preliminary AROT. For this purpose, the ADS-B data points of each flight were sorted chronologically by timestamp. We then iterated through the sorted data and assigned each data point to one of the defined polygons based on its position. The iteration continued until ten consecutive data points of the same flight were located within the *runway-vacated polygon*, while being outside the runway polygon and below a gealtitude of 2000 ft. Once this condition was met, the iteration was terminated under the assumption that the aircraft had vacated the runway. This approach was chosen to prevent delayed ADS-B messages from artificially increasing the calculated AROT. The altitude

threshold of 2000 ft ensured that the iteration did not terminate prematurely, for example in the case of a go-around, when an aircraft could temporarily enter the *runway-vacated polygon* during the manoeuvre. The preliminary AROT was then computed as the time difference between the first and last data point located within the runway polygon.

We validated the preliminary AROT using a set of rule-based algorithms. If the calculated AROT was shorter than 40 s, we repeated the determination process without applying the *runway-vacated polygon*, as such short durations suggested that the iteration may have been terminated prematurely due to noisy ADS-B data. When the preliminary AROT exceeded 150 s, we analysed the flight for a potential go-around using the `go_around()` function of the *traffic* library. If a go-around was detected, or if at least one data point above 1800 ft was located within the go-around polygon, we examined the time differences between successive data points inside the runway polygon. In cases where the time difference between two consecutive data points within the runway polygon exceeded 150 s, we removed all preceding points from the polygon and recalculated the preliminary AROT. If no such gap was found, we excluded the flight from the dataset, as it was assumed that a go-around had occurred and the aircraft subsequently landed on a different runway or diverted.

2.1.3 Determination of Rapid Exit Taxiway

After completing the AROT validation process, we determined the RET used by each aircraft after landing. For this purpose, we defined a polygon above each RET that directly connected to the runway polygon. Accordingly, for Zurich Airport's runway 14, we defined three different RET polygons corresponding to exit taxiways H1, H2, and H3, as illustrated in Figure 4.

Starting from the last data point recorded within the runway polygon, we evaluated up to ten subsequent data points and assigned them to the corresponding RET polygons. The RET with the highest number of assigned points was identified as the exit used by the aircraft. Flights for which the majority of these data points did not fall within any RET polygon were excluded from the dataset.

2.1.4 Time-Variant Features

After determining both the AROT and the RET, we created a *parquet* file containing trajectory snippets for all remaining flights. To define these snippets, we set the prediction point at a distance of 4 NM to the threshold of runway 14 and constructed each trajectory snippet as the last ten data points recorded before the aircraft crossed this point. This choice was made in order to remain consistent with one of the prediction points used by Martínez et al. [6], who also considered a distance of 4 NM to the runway threshold. As shown in Figure 2, each snippet includes the following features: groundspeed, gealtitude, and vertical rate. For each feature, we applied predefined thresholds to remove values we considered unrealistic. Specifically, we excluded data points from the snippets where the gealtitude is larger than 5000 ft or the groundspeed exceeds 240 kt in order to minimise the impact of outliers on the prediction model. Finally, we removed all flights whose snippets did not contain at least ten valid data points.

2.1.5 Time-Invariant Features

To collect the time-invariant features in a structured format, we created a *Pandas* table containing the AROT and RET of each flight, together with additional flight-related features [14]. In the first step, we added the AROT, RET, aircraft type, and landing time, defined as the timestamp of the first data point within the runway polygon, to the table. Since the number of observations was low for several aircraft types, we defined a minimum frequency threshold of 50. Aircraft types with fewer observations than this threshold were grouped into broader categories, such as *Rare Light* or *Rare Medium*, based on their ICAO Weight Turbulence Category. We also added the time of prediction, defined as the timestamp of the last data point in the trajectory snippet. To capture the cyclic nature

of time, we applied a cyclic encoding scheme to the time of prediction, transforming the day of the week and the month of the year into sine and cosine values ranging from 0 to 1. In addition, we added the groundspeed, gealtitude, and vertical rate of the last data point to the table, making them available to the RET model, whereas the AROT model relied solely on the trajectory snippet features.

In the next step, we added missing aircraft type data, weather information, and aircraft characteristics data to the dataset. The missing aircraft types were retrieved from a manually created database containing the ICAO24 address and aircraft type. We added precipitation, temperature, wind speed, and wind direction from historical *MeteoSwiss* weather data recorded at the Kloten station [15]. Furthermore, we imported visibility information from historical METAR reports using the *traffic* library [13]. For each flight, the weather data were matched based on the closest timestamp prior to the time of prediction. Visibility values were converted into ordinal classes to handle text-based entries such as “greater than 10000 meters”. Given the 30-minute sampling interval of METAR data and the 10-minute interval of the *MeteoSwiss* data, we treated all weather features as time-invariant. Using the FAA Aircraft Characteristics Database [16], we further supplemented the dataset with the ICAO Wake Turbulence Category (WTC) and the Maximum Allowable Landing Weight (MALW), while missing values were filled using a manually created database containing aircraft type, WTC, and MALW. To incorporate information on the arrival stand, we divided Zurich Airport into four stand regions—North, East, South, and West—as shown in Figure 5. This simplified division reduced the number of possible stand assignments while preserving relevant spatial information and allowed stand regions to be assigned even for flights whose ADS-B data ended before reaching the actual stand. For each flight, the last five data points were compared with these regions, and the region containing the majority of points was assigned as the stand region; if most points were outside all defined regions, the stand region was marked as unknown.

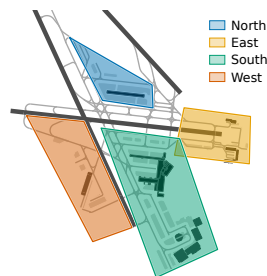


Figure 5. Visualization of Polygons for Stand Regions at Zurich Airport

Further features were engineered from the data available in the *Pandas* table. Using wind data and runway orientation, we calculated the headwind and crosswind components. The ICAO airline designator was extracted from the first three letters of each callsign. To ensure statistical relevance, we applied a minimum occurrence threshold of 50 for unique airline identification, grouping designators below this threshold into rare categories based on their associated stand region, such as *Rare North* or *Rare South*.

In a final step, we checked all features for missing values and outliers. Entries affected by these issues were either removed or replaced with appropriate values to ensure data consistency. This procedure was necessary since the AROT prediction model required scaled inputs and could not process missing data. To mitigate noise arising from cases in which AROT was not a limiting factor, we adopted the same data-filtering approach as Stempfel et al. [8] used in their study. Therefore, we removed all flights for which the time separation at the runway threshold from the following aircraft

exceeded 180 s for *Heavy* and *Super* aircraft types, and 120 s for all other categories. In addition, we excluded all entries of flights that vacated via rapid exit taxiway H3, since this exit was strongly under-represented and rarely relevant for cases where flights were constrained by AROT. The final dataset comprised 55,638 samples. It was then randomly divided into two equally sized parts using the `train_test_split` function from the *scikit-learn* library [17]. The first part, referred to as the RET dataset, was used to train and validate the RET prediction model, while the second part, referred to as the AROT dataset, was used to test the RET model and to train, validate, and test the AROT prediction model.

2.2 RET Prediction Model

To predict the RET selected by an aircraft upon landing, we used a Light Gradient Boosting Machine (LightGBM) model [18]. The prediction task was formulated as a multiclass classification problem. Since a strong correlation between the selected RET and the AROT was expected, the model was configured to output a probability distribution across all RET classes, representing the likelihood of each RET being selected. This probability distribution was later used as an input feature for the subsequent AROT prediction model (see Section 2.3). For the RET prediction model, we selected the logarithmic loss (log loss) as the evaluation metric, as it measured the accuracy of a classification model that outputs probability values between 0 and 1. A lower log loss indicates better-calibrated and more confident probability estimates. To further evaluate model performance, we defined the predicted RET as the class with the highest assigned probability.

To train the model, we divided the RET dataset into a training and validation set using an 80:20 ratio. The training set was used to fit the model, while the validation set served to monitor performance and prevent overfitting. To identify the optimal hyperparameter configuration, we employed the *Optuna* framework for automated Bayesian optimization based on a Tree-structured Parzen Estimator [19]. The feature set was defined heuristically based on domain knowledge and exploratory analysis. The configuration yielding the lowest log loss was selected for the final RET prediction model. An overview of the features used in this final model is provided in Figure 2. We then evaluated the model performance on the AROT dataset and compared it to a naive baseline. For this baseline, we determined the most frequently used RET per aircraft type from the training set and applied this mapping to the AROT dataset. After completing the evaluation, we generated predictions on the AROT dataset and appended the predicted probabilities for each RET class to this dataset.

2.3 AROT Prediction Model

After completing the development and evaluation of the RET prediction model, we developed the AROT prediction model. This sequential setup was required because the AROT model used the predicted RET probabilities as input features, and simultaneous optimisation of both models would have introduced data leakage. In contrast to the RET prediction model, the AROT model considered both time-invariant features such as airline, and time-variant features from the trajectory snippets, including groundspeed, gealtitude and vertical rate. To handle this structure, we implemented a fully connected neural network with two separate input branches using the *Keras* library [20], as this architecture outperformed a single-input architecture during initial prototyping. Figure 6 illustrates the structure of the AROT prediction model. The time-invariant branch passed its inputs through a hidden layer with 128 neurons, while the time-variant branch flattened the input and processed it through two hidden layers with 128 and 64 neurons, respectively. The outputs of both branches were then concatenated and passed through a dropout layer with a rate of 15 % before being fed into a single-neuron output layer for regression. All hidden layers employed the ReLU activation function. Early stopping was implemented to terminate training once the validation loss failed to improve for 15 consecutive epochs, thereby preventing overfitting and unnecessary computation. Additionally, a `ReduceLROnPlateau` callback was applied to adapt the learning rate when no further

improvement was observed, ensuring a stable and efficient convergence process.

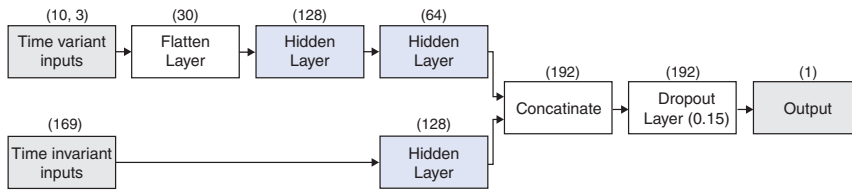


Figure 6. Dual-Input Architecture of the AROT Prediction Model

Before training, we divided the AROT dataset and the corresponding trajectory dataset into training, validation, and test sets using a 70:15:15 ratio. Because neural networks cannot directly process missing values or categorical variables, we removed all remaining entries with missing data and applied one-hot encoding to the categorical features. All continuous input variables, except those that were cyclically or one-hot encoded, were then normalized to a 0–1 range to ensure that each feature contributed equally to the training process and to promote stable and efficient gradient-based optimization. To prevent data leakage, the normalization parameters were derived exclusively from the training set and subsequently applied to the validation and test sets. We applied the same procedure as described for the RET prediction model in Section 2.2 to define the feature set, while the hyperparameters were tuned using a random search. Figure 2 provides an overview of all features used in the final AROT prediction model. The evaluation metric was the RMSE, defined as the square root of the average squared difference between the predicted and observed values. To assess the contribution of the trajectory snippets, we additionally trained a single-input fully connected neural network using only time-invariant features together with the corresponding values of groundspeed, geoaltitude, and vertical rate at the prediction point. The architecture of this baseline model was determined using the same random search procedure, resulting in two hidden layers with 128 and 256 neurons, respectively, followed by a dropout layer. All other hyperparameters were kept identical to those of the dual-input model. Finally, we conducted a permutation feature importance analysis to qualitatively assess the influence of each feature on the AROT prediction model.

3. Results

The following section presents the results of our study and is divided into two parts: Section 3.1 summarises the findings of the RET prediction model, while Section 3.2 reports the results of the AROT prediction model.

3.1 RET Prediction Model

Table 1 presents the classification report of the final RET prediction model and the baseline on the test dataset. The columns show the precision, recall, F1-score, and support for each predicted RET class. Precision indicates the proportion of correct positive predictions relative to all predicted positives, while recall measures the proportion of actual positives correctly identified by the model. The F1-score is the harmonic mean of precision and recall, combining them into a single performance measure. Support refers to the number of true instances for each class. The bottom rows of the table include the macro average and weighted average across all classes. The model achieved a log loss of 0.443 and an accuracy of 79.4%. The baseline achieved an accuracy of 77.8%.

Figure 7 depicts the calibration curves of the final RET prediction model on the test dataset. The horizontal axis represents the predicted probability for each RET class, while the vertical axis indicates the corresponding observed frequency. Each curve shows how well the predicted probabilities

Table 1. Classification Report of the Final RET Prediction Model and Baseline on the Test Dataset

Class	LightGBM				Baseline			
	Precision	Recall	F1	Support	Precision	Recall	F1	Support
RET H1	0.80	0.95	0.87	19649	0.77	0.98	0.86	19649
RET H2	0.77	0.43	0.55	8169	0.89	0.28	0.42	8169
Macro Avg	0.78	0.69	0.71	27818	0.83	0.63	0.64	27818
Weighted Avg	0.79	0.79	0.77	27818	0.80	0.78	0.73	27818

align with the actual outcomes for the respective RET class. The dashed red line represents perfect calibration, where predicted probabilities match the observed frequencies exactly.

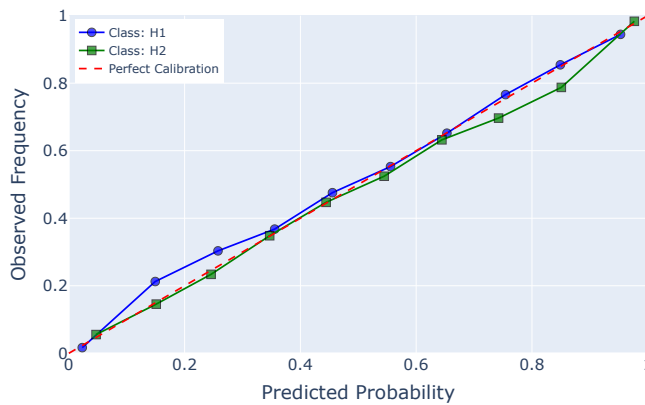


Figure 7. Calibration Curves of the RET Prediction Model on the Test Dataset

3.2 AROT Prediction Model

Table 2 presents the performance metrics achieved by the final AROT prediction model on the test set, compared to a model without trajectory snippets. The table reports the RMSE, the MAE, the R^2 , the median absolute error, and the 90th percentile error. MAE represents the average absolute difference between predicted and actual values, while R^2 indicates the proportion of variance in the target variable explained by the model.

Table 2. Performance Metrics of the Final AROT Model and Single-Input Baseline on the Test Dataset

Metric	With Trajectory Snippets	Without Trajectory Snippets
Mean Absolute Error (MAE) [s]	3.95	4.00
Median Absolute Error [s]	3.34	3.36
Root Mean Square Error (RMSE) [s]	5.01	5.10
90th Percentile Absolute Error [s]	7.97	8.04
R^2 Score	0.32	0.30

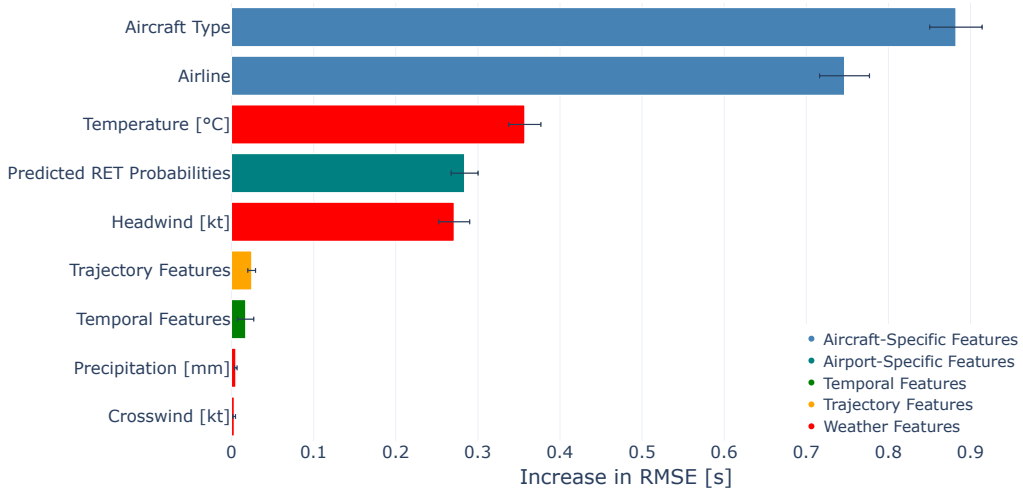


Figure 8. Permutation-Based Feature Importance for RET Prediction Based on 100 Repetitions

Figure 8 presents the permutation-based feature importance for predicting AROT using the final prediction model. The horizontal axis shows the increase in RMSE when each feature is randomly permuted, while the vertical axis lists the features. For visualization purposes, trajectory and temporal features are displayed as aggregated categories, although permutation is performed on the individual features. The bars indicate the resulting performance degradation, and the error bars reflect uncertainty based on variability across permutations. The bar colors represent feature categories. Aircraft type shows the highest increase in RMSE, while features such as crosswind and precipitation have minimal impact on log loss.

4. Discussion

4.1 RET Prediction Model

The final RET prediction model achieves an overall accuracy of 79.4 %, correctly predicting the RET used in almost four out of five cases. However, a closer inspection of Table 1 reveals substantial differences in predictive performance between the two RETs installed at runway 14 at Zurich Airport that were considered in this study. While the model reliably predicts the majority RET H1, reaching a precision of 0.80 and a recall of 0.95, its performance for the minority RET H2 is considerably weaker, with a precision of 0.77 and a recall of only 0.43. This discrepancy illustrates the model's difficulty in correctly identifying H2 cases, which is also reflected in the calibration curve in Figure 7. Although the predicted probabilities follow the perfect calibration relatively closely overall, the curves deviate at the extremes: at a predicted probability of 0.2, H1 is under-confident and, consequently, H2 at a corresponding predicted probability of 0.8 is overconfident. Compared to the naive baseline, the model nevertheless achieves a slightly higher overall accuracy of 79.4 % compared to 77.8 %. For RET H1, the predictive performance is similar for both approaches, with F1-scores of 0.87 for the model and 0.86 for the baseline. For RET H2, however, the model performs better, achieving an F1-score of 0.55 compared to 0.42 for the baseline, mainly due to a substantially higher recall of 0.43 instead of 0.28.

One potential explanation for the observed differences in predictive performance between RET H1 and H2 lies in the distance of the RETs relative to the runway threshold as well as their geometry. Specifically, the first RET, H1, is located at a distance of approximately 2,300 m from the runway

threshold. At such a distance, most arriving aircraft are able to decelerate sufficiently to vacate via H1. Consequently, there are fewer operational or physical reasons to vacate via a RET positioned further from the runway threshold, such as H2. This reasoning is supported by the Aeronautical Information Publication of Zurich Airport, which states that arriving aircraft on runway 14 are generally advised to vacate the runway via RET H1 whenever possible, except for aircraft in the wake turbulence category *HEAVY* or when instructed otherwise by ATC [21]. As a result, the influence of human behaviour on RET selection increases for taxiways located further from the runway threshold of runway 14. Human behaviour is inherently more difficult to predict, as it depends on both individual and situational factors. In the context of RET selection, it comprises two main components: the actions of the cockpit crew and the decisions of ATC. While the behaviour of the cockpit crew should be reflected in the prediction of RET selection, the decisions of ATC should not. Our model is designed to support ATC operations and not to anticipate future ATC instructions. Including cases in which ATC influences the RET selection, for instance by advising or permitting an aircraft to vacate via RET H2, would introduce noise and reduce the model's ability to learn meaningful patterns. Furthermore, such cases would distort the evaluation metrics, as they do not represent the model's predictive performance in operationally relevant scenarios. Considering the applied filtering thresholds, the observed average AROT of approximately 55 s, and the observed time separations between successive arrivals shown in Figure 1, we conclude that a considerable number of irrelevant cases remain in our dataset. These cases likely continue to affect both the model's performance and the interpretation of its evaluation metrics. To address this limitation, a more refined filtering approach would be required, for example by incorporating ATC communication logs to identify cases in which controller interaction may have influenced the RET selection.

Compared with Woo et al. [5], the performance of our RET model is somewhat lower than the results reported in their study. They developed a RET prediction model for runway 02L at Singapore Changi Airport, achieving an accuracy of 80.76 %, a precision of 86.99 %, a recall of 80.76 %, and an F1-score of 82.91 %. In comparison, our model achieves an accuracy of 79.4 %, a weighted precision of 79 %, a weighted recall of 79 %, and a weighted F1-score of 77 %. However, the comparability of these results is limited due to fundamental differences in runway layout and airport operations between Singapore Changi and Zurich Airport. Selecting RETs located closer to the runway threshold of runway 02L at Singapore Changi Airport often results in shorter taxi times, as aircraft landing on this runway typically need to taxi back in the opposite direction to reach their stands. Because this runway-taxiway configuration naturally encourages pilots to vacate the runway earlier, braking pressure during rollout tends to be higher. Consequently, the variability of human behaviour in RET selection is reduced, since there is no trade-off between competing objectives such as braking comfort and taxi efficiency. In contrast, on runway 14 at Zurich Airport, taxiing continues in the direction of the landing rollout. As a result, selecting a RET that is located farther down the runway can reduce taxi time. This introduces a trade-off between vacating the runway early and staying on it longer to minimise taxi time, thereby increasing the influence of individual pilot behaviour on RET selection.

4.2 AROT Prediction Model

In most cases, the prediction errors of the final AROT prediction model presented in this study are small relative to the observed time buffers between two consecutive landings on runway 14 shown in Figure 1. This finding is supported by the model's MAE of 3.95 s and median absolute error of 3.34 s. Nevertheless, our AROT prediction model occasionally exhibits relatively large errors in certain situations, as indicated by a RMSE of 5.01 s and a 90th percentile error of 7.97 s. These deviations can be attributed to several factors. Firstly, with regard to the dataset, the AROT prediction model is affected by the same limitations as the RET prediction model. In cases where AROT was not a limiting factor, ATC may have advised or approved the use of a later RET and thereby influenced the

observed AROT. As mentioned previously, such cases should not be included, as they do not reflect operationally relevant situations and introduce noise that affects both the model's performance and the interpretation of its evaluation metrics. Secondly, AROT can also be significantly influenced by the actual touchdown point of the aircraft, braking strategy of the pilots, and runway exit speed. For instance, an early touchdown near the runway threshold combined with a strong initial braking phase and a low exit speed could increase AROT considerably. At the prediction point, none of this information is available, and the model therefore cannot account for this variability. This constraint is partly reflected in the R^2 value of 0.32, indicating that the selected features explain roughly one third of the total variance of AROT, while the remaining variance is likely driven by unobserved factors not available at the prediction point.

Comparisons with previous studies show that our AROT model slightly outperforms existing models across all key evaluation metrics. Jun, Wei, and Alam [7] obtained MAE values between 4.35 s and 4.74 s and RMSE values from 5.76 s to 6.10 s for models predicting AROT for runway 02L at Singapore Changi Airport. Martinez et al. [4] reported an MAE of 8 s for runway 34 at Vienna Airport, with 80 % of flights predicted within 14 s, while Martínez et al. [6] achieved about 90 % within 10 s at Barcelona El Prat. In comparison, our model reaches an MAE of 3.95 s, an RMSE of 5.01 s, and 90 % of predictions within 7.97 s. Although these results are favourable, their comparability with previous studies is limited. Our approach applied a filtering method following Stempfeler et al. [8] to remove flights in which AROT was not a limiting factor, whereas Jun, Wei, and Alam excluded outliers beyond three standard deviations, and both Martinez et al. and Martínez et al. did not apply any filtering. Differences in prediction distance, traffic composition, and runway configuration further restrict direct comparison. Moreover, both Jun, Wei, and Alam and Martinez et al. relied exclusively on time-invariant features available at the prediction point, whereas Martínez et al. incorporated time-variant information by fitting polynomial coefficients to represent the descent phase, while our model used trajectory snippets. Finally, we employed a random train-test split, similar to the train-test and cross-validation strategies used by Jun, Wei, and Alam and the stratified k-fold cross-validation applied by Martinez et al. This approach may have allowed a certain degree of temporal data leakage due to correlated operational conditions, whereas the explicit temporal split applied by Martínez et al. would more rigorously prevent such leakage and better reflect operational forecasting scenarios. Consequently, the results of our model appear plausible and in line with findings from previous research, although direct performance comparison remains limited due to methodological and contextual differences.

The results of the permutation feature importance depicted in Figure 8 confirm that the features derived from the trajectory snippets are utilised by the model. However, their overall contribution to model performance appears to be minimal. This is further supported by the comparison with the separately trained model without trajectory snippets. While the trajectory snippet model achieves marginally lower prediction errors than the model without trajectory snippets, the difference remains small and cannot be considered meaningful given the applied methodology. We suspect that this is mainly due to the limited additional information provided by the trajectory snippets at the chosen prediction point, which is located at a distance of 4 NM from the runway threshold. At this distance, the aircraft is typically fully established on the ILS or in a stable visual approach. Consequently, the variability in groundspeed, gealtitude, and vertical rate is limited. In addition, pilots still have sufficient time to correct deviations during this phase. Overall, we expect the contribution of trajectory snippets to increase as the prediction point moves closer to the runway threshold. From an operational perspective, we assume that the practical value of a model with a prediction point located 4 NM or closer to the runway threshold is rather limited, as aircraft separation has already been established at this stage. Therefore, a model operating at such a distance from the threshold is likely to be most useful for supporting ATC decision-making in determining whether a go-around is required. A model intended to support ATC in establishing optimal spacing, on the other hand,

would need to predict AROT several minutes prior to landing. However, at such longer prediction horizons, trajectory snippets are likely to have only a minor impact on model performance. In addition, the applied filtering approach may limit operational applicability. Removing data points can result in snippets with fewer than ten points, in which case the model cannot produce a prediction. Threshold exceedances could instead be handled through simple capping or imputation, provided that the same strategy is applied during model training.

With regard to generalisability, we consider both the method used to predict RET usage from ADS-B data and the method applied to predict AROT using ADS-B trajectory snippets to be transferable to other runways and airports. However, both methods rely on adequate ground coverage; if coverage is insufficient, it may not be possible to determine AROT or the selected RET accurately. For airports with limited or no available data, it may be of interest to apply a fully trained model from another aerodrome. Nevertheless, we do not expect a model trained on a specific runway to perform well on a different one, as both AROT and RET selection are strongly influenced by runway characteristics and the overall airport layout. In this context, Nguyen et al. [9] showed that, for airports with limited AROT data, model performance can be improved by training on data from one or multiple other airports and including the target airport's data as only a small fraction of the overall dataset. In their approach, the runways were represented through numerical equivalent features, which allowed the model to generalise across different airports. In certain cases, this method achieved better results than models trained solely on the limited data of the target airport. In line with this, it is also advisable to replace all remaining categorical features with numerical equivalents, as done by Nguyen et al. [9], in order to increase robustness and improve the utility of the model in cold-start deployment scenarios. For RET prediction, on the other hand, it is unlikely that a specific RET can be predicted using a model trained on another runway. Instead, one would need to limit the prediction to whether an arriving aircraft vacates via a procedural exit or not. This could be achieved by including numerical features that describe the number and location of procedural exits on a given runway, for instance by defining features that represent the minimum and maximum distance of procedural exits from the runway threshold. Ultimately, the practical value of a generalised prediction model would depend on whether the achieved performance is sufficient for its intended operational use.

5. Conclusion and Outlook

This study investigated the potential of ADS-B trajectory data from the OpenSky Network to predict RET selection and AROT at Zurich Airport. A LightGBM model was trained to predict the RET used by arriving aircraft on runway 14, and a fully connected neural network with separate input branches for time-invariant features and trajectory snippets was developed to predict AROT. The RET model achieved a weighted accuracy of 79.4 %, while the AROT model reached a MAE of 3.95 s and a RMSE of 5.01 s. These results demonstrate that both RET and AROT can be predicted using openly available ADS-B data with a performance level comparable to studies based on proprietary A-SMGCS or radar data.

While the RET model performed reliably for the majority exit H1, its performance was lower for H2. This difference can be explained by the relative positions of the exits and the operational procedures published in the Aeronautical Information Publication of Zurich Airport, which advise aircraft to vacate via H1 whenever possible. Consequently, the variability of human behaviour and ATC interaction is greater for H2, making these cases more difficult to predict. Despite this, the model still outperforms a naive baseline, particularly for RET H2. Regarding the AROT model, it produced small errors for most flights, although larger deviations occurred in situations influenced by unobserved factors such as touchdown point, braking behaviour, and runway exit speed. The coefficient of determination of 0.32 confirms that part of the AROT variance is explained by variables not available at the prediction point. Furthermore, a comparison with a single-input model trained without

trajectory snippets showed no meaningful difference in predictive performance, indicating that trajectory snippets provide only limited additional information at the chosen prediction point of 4 NM. Nevertheless, both models were likely affected by remaining irrelevant cases in the dataset. Despite the applied filtering procedure, flights in which AROT was not a limiting factor or in which ATC interventions influenced the exit selection were not fully removed. These cases may have affected both model training and performance evaluation.

This study has shown that the developed prediction models provide practical operational value. By delivering reliable estimates of AROT and RET selection based solely on openly available ADS-B data, the models can support air traffic controllers in adjusting arrival spacing and sequencing during approach operations. In this way, our models potentially enable a more efficient utilisation of existing runway infrastructure and contribute to an increase in arrival throughput without the need for costly physical expansion.

Future work should examine the effect of the trajectory snippet configuration on model performance, including snippet length, sampling rate, number of snippets, and their position within the descent phase. Moreover, RET prediction could be extended to runways with exits on both sides to evaluate the method's applicability to more complex layouts. Finally, the generalisability of both models should be explored by training and testing them on data from other runways or airports. In this context, combining data from multiple airports and incorporating numerical runway descriptors and other property-based features, as proposed by Nguyen et al. [9], may improve model transferability and robustness, particularly in cold-start scenarios with limited local data.

Acknowledgement

The authors acknowledge the contributions of two/three reviewers that greatly enhanced the value of this study. No potential conflict of interest was reported by the authors. No funding was received for this research.

Author Contributions

- Kevin Hänggi: Conceptualization, Methodology, Data Curation, Software, Validation, Visualization, Writing (Original Draft and Editing)
- Jeremy Wilde: Writing (Review)
- Manuel Waltert: Conceptualization, Writing (Editing, Review), Project Administration

Open Data Statement

The software code used to download the OSN-data employed in this study is available on the following repository: https://github.com/hanvkev/osn25_AROT_RET.

Reproducibility Statement

The software code used to generate the results presented in this paper is available on the following repository: https://github.com/hanvkev/osn25_AROT_RET.

References

- [1] EUROCONTROL. *EUROCONTROL aviation long-term outlook: Flights and CO₂ emissions forecast 2024–2050*. EUROCONTROL, Dec. 2024. URL: <https://www.eurocontrol.int/publication/eurocontrol-forecast-2024-2050>.

- [2] Floris Herrema, Ricky Curran, Sander Hartjes, Mohamed Ellejmi, Steven Bancroft, and Michael Schultz. “A machine learning model to predict runway exit at Vienna Airport”. In: *Transportation Research Part E: Logistics and Transportation Review* 131 (2019), pp. 329–342. ISSN: 13665545. doi: 10.1016/j.tre.2019.10.002.
- [3] EUROCONTROL. *ROCAT solution - Optimised runway occupancy time spacings for arrivals*. Apr. 19, 2023. URL: <https://www.eurocontrol.int/publication/optimised-runway-occupancy-time-spacings-arrivals> (visited on 10/24/2024).
- [4] Dario Martinez, Seddik Belkoura, Samuel Cristóbal, Innaxis, and Floris Herrema. “A boosted tree framework for runway occupancy and exit prediction”. In: *Proceedings of the 8th SESAR Innovation Days*. 3–7 September 2018. Salzburg, Austria: SESAR Joint Undertaking, Sept. 2018.
- [5] Chuan Jie Woo, Sim Kuan Goh, Sameer Alam, Md Meftahul Ferdous, and Mohamed Ellejmi. “A runway exit prediction model with visually explainable machine decisions”. In: *Proceedings of the 10th International Conference on Research in Air Transportation (ICRAT)*. Tampa, FL, USA: ICRAT, June 2022, pp. 1–9.
- [6] Miguel García Martínez, Javier García Moreno, Rubén González Sendino, and Álvaro Rodríguez Sanz. “Resilient arrival runway occupancy time prediction for decision-making tool in Barcelona (LEBL) Airport”. In: *Proceedings of the 9th International Conference on Research in Air Transportation (ICRAT)*. Tampa, FL, USA, June 2020.
- [7] Lim Zhi Jun, Chow Hong Wei, and Sameer Alam. “Data-driven runway occupancy time prediction using decision trees”. In: *Proceedings of the 2021 IEEE/AIAA 40th Digital Avionics Systems Conference (DASC 2021)*. San Antonio, TX, USA: IEEE, 2021, pp. 1–9. doi: 10.1109/DASC52595.2021.9594365.
- [8] G. Stempf, I. De Visscher, M. Ellejmi, V. Brossard, A. Bonnefoy, and V. Trève. “Applying machine learning modelling to enhance runway throughput at a major European airport”. In: *IOP Conference Series: Materials Science and Engineering* 1024.1 (2021), p. 012106. ISSN: 1757-8981, 1757-899X. doi: 10.1088/1757-899X/1024/1/012106.
- [9] An-Dan Nguyen, Duc-Thinh Pham, Nimrod Lilith, and Sameer Alam. “Model generalization in arrival runway occupancy time prediction by feature equivalences”. In: *Proceedings of the 9th International Conference for Research in Air Transportation (ICRAT 2020)*. 2020. doi: 10.48550/arXiv.2201.11654. URL: <https://arxiv.org/abs/2201.11654>.
- [10] Navid Mirmohammadsadeghi and Antonio Trani. “Prediction of runway occupancy time and runway exit distance with feedforward neural networks”. In: *Proceedings of the 2019 Aviation Technology, Integration, and Operations Conference (ATIOC)*. Portland, OR, USA: AIAA, 2019. doi: 10.2514/6.2019-3044.
- [11] Matthias Schäfer, Martin Strohmeier, Vincent Lenders, Ivan Martinovic, and Matthias Wilhelm. “Bringing up OpenSky: A large-scale ADS-B sensor network for research”. In: *IPSN-14 Proceedings of the 13th International Symposium on Information Processing in Sensor Networks*. IEEE, 2014, pp. 83–94.
- [12] Junzi Sun. *The 1090 megahertz riddle: A guide to decoding Mode S and ADS-B signals*. 2nd ed. TU Delft OPEN Publishing, 2021. ISBN: 978-94-6366-402-8. doi: 10.34641/MG.11.
- [13] Xavier Olive. “traffic, a toolbox for processing and analysing air traffic data”. In: *Journal of Open Source Software* 4.39 (2019), p. 1518. ISSN: 2475-9066. doi: 10.21105/joss.01518.
- [14] Wes McKinney. “Data structures for statistical computing in python”. In: *Python in Science Conference*. Austin, Texas, 2010, pp. 56–61. doi: 10.25080/Majora-92bf1922-00a. (Visited on 06/04/2025).
- [15] Federal Office of Meteorology and Climatology (MeteoSwiss). *Open data – MeteoSwiss*. May 2025. URL: <https://www.meteoswiss.admin.ch/services-and-publications/service/open-data.html> (visited on 08/28/2025).
- [16] Federal Aviation Administration. *FAA aircraft characteristics database*. Oct. 2023. URL: https://www.faa.gov/airports/engineering/aircraft_char_database/ (visited on 08/28/2025).

- [17] Fabian Pedregosa et al. “Scikit-learn: machine learning in python”. Version 4. In: (2012). DOI: 10.48550/ARXIV.1201.0490.
- [18] Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu. “LightGBM: A highly efficient gradient boosting decision tree”. In: *Advances in Neural Information Processing Systems*. Ed. by I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett. Vol. 30. Curran Associates, Inc., 2017, pp. 3146–3154.
- [19] Takuya Akiba, Shotaro Sano, Toshihiko Yanase, Takeru Ohta, and Masanori Koyama. “Optuna: A next-generation hyperparameter optimization framework”. In: *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 2019.
- [20] François Chollet et al. *Keras*. 2015. URL: <https://keras.io>.
- [21] Skyguide. *Aeronautical information publication Switzerland: Zurich Airport (LSZH)*. Bern, Switzerland, 2025.