Predicting Estimated Time of Arrival for Commercial Flights

Samet Ayhan∗
University of Maryland
College Park, Maryland
sayhan@cs.umd.edu

Pablo Costas
Boeing Research & Technology
Europe
Madrid, Spain
pablo.costas@boeing.com

Hanan Samet
University of Maryland
College Park, Maryland
hjs@cs.umd.edu

ABSTRACT
Unprecedented growth is expected globally in commercial air traffic over the next ten years. To accommodate this increase in volume, a new concept of operations has been implemented in the context of the Next Generation Air Transportation System (NextGen) in the USA and the Single European Sky ATM Research (SESAR) in Europe. However, both of the systems approach airspace capacity and efficiency deterministically, failing to account for external operational circumstances which can directly affect the aircraft’s actual flight profile. A major factor in increased airspace efficiency and capacity is accurate prediction of Estimated Time of Arrival (ETA) for commercial flights, which can be a challenging task due to a non-deterministic nature of environmental factors, and air traffic. Inaccurate prediction of ETA can cause potential safety risks and loss of resources for Air Navigation Service Providers (ANSP), airlines and passengers. In this paper, we present a novel ETA Prediction System for commercial flights. The system learns from historical trajectories and uses their pertinent 3D grid points to collect key features such as weather parameters, air traffic, and airport data along the potential flight path. The features are fed into various regression models and a Recurrent Neural Network (RNN) and the best performing models with the most accurate ETA predictions are compared with the ETAs currently operational by the European ANSP, EUROCONTROL. Evaluations on an extensive set of real trajectory, weather, and airport data in Europe verify that our prediction system generates more accurate ETAs with a far smaller standard deviation than those of EUROCONTROL. This translates to smaller prediction windows of flight arrival times, thereby enabling airlines to make more cost-effective ground resource allocation and ANSPs to make more efficient flight schedules.

CCS CONCEPTS
• Mathematics of computing → Multivariate statistics; • Information systems → Data analytics; • Applied computing → Aerospace;

∗ Senior Engineer at Boeing Research & Technology.
usually begin by estimating the flight trajectory, which includes the lateral flight path together with altitude and speed profiles, and then proceed to calculating the time required to fulfill the predicted trajectory. These models by themselves fail to account for external operational circumstances such as weather phenomena, airspace sector densities, and airport congestion, which can directly affect the aircraft’s actual flight profile. Note that decreasing the average delay per flight by one minute could save millions of dollars in annual crew costs and fuel savings for a mid-sized airline. Hence, unlike traditional methods, we propose a machine learning based systematic approach to address the ETA prediction problem. We make a considerable effort in feature engineering to use a richer set of features including general information about flights as well as weather, air traffic, and airport data for more accurate modeling and prediction. Our prediction system is built upon the concepts presented in the Aircraft Trajectory Prediction System [5] in which airspace is considered as a set of data cubes around grid points as part of the 3D reference grid network.

The paper contains three main contributions:

- We propose an ETA Prediction System for commercial flights. Unlike other systems that collect and use features only for the arrival airport, we use a richer set of features along the potential route, such as weather parameters and air traffic data in addition to those that are particular to the arrival airport. Our feature construction process generates an extensive set of multidimensional time series data which goes through Time Series Clustering with Dynamic Time Warping (DTW) to generate a single set of representative features at each time instance.

- We present algorithms for two major features that highly contribute to the accurate ETA prediction: airspace sector congestion rate, and airport congestion rate.

- We perform a comparative analysis using a number of regression models and an RNN and present their performances. Using the best performing model, we compare our results with those generated by the EUROCONTROL. Our experiments on real trajectory, weather, air traffic and airport data verify that our system outperforms EUROCONTROL’s ETA prediction by offering not only a higher accuracy but also a far smaller standard deviation, resulting in smaller prediction windows of flight arrival times.

The proposed system can be used by the ANSPs and airlines for more accurate scheduling and resource management resulting in improvement in the four ATM key performance areas; safety, capacity, efficiency, and environmental impact. The rest of the paper is organized as follows. Section 2 reviews related work. Section 3 describes the data used in this study followed by Section 4 where we present a proper set of features selected for ETA prediction via feature engineering. Section 5 states the problem and reviews two top-ranking boosting methods, while Section 6 presents our experimental evaluation. Section 7 draws concluding remarks.

2 RELATED WORK

There has been much work and an abundant literature on predicting estimated rate of arrival for humans [23] and ETA for various transportation vehicles on land (road and rail) [9, 25] and water [29], in addition to air. Traditional methods to ETA prediction for air transportation, particularly in the case of commercial flights use a deterministic approach by heavily relying on aircraft performance models along with either parametric or physics-based trajectory models. They usually begin by computing the flight trajectory, which includes the lateral flight path together with altitude and speed profiles, and then proceed to calculate the time required to fulfill the predicted trajectory [7, 21, 22, 24, 42]. However, the time of arrival at a fixed point on the ground is dependent not only on the airspeed the aircraft will fly, but also on the winds, temperatures, air traffic along the route and congestion at the arrival airport. Therefore, if the atmospheric, airspace and airport traffic data is uncertain, the reference trajectory may not have been laid out correctly with respect to the real world situation, yielding inaccurate prediction of estimated time-of-arrival for the specified airport. Probabilistic methods usually develop stochastic linear models of aircraft motion to estimate future aircraft positions. They model aircraft trajectories as Discrete-Time Stochastic Hybrid Linear Systems in which aircraft path decisions form hybrid modes which then inform mode transitions between turning and straight flight modes [17, 31, 43].

Unlike traditional methods, we propose a machine learning based systematic approach in which general information about the flight as well as weather and air traffic data are all taken into consideration. Our objective is to learn from historical data, possibly stored in an aviation data warehouse [3], discover patterns and generate a set of features to build a model for accurate ETA prediction. There are only a few papers following the machine learning approach to the ETA prediction problem for air transportation in the literature [15, 20, 40].

In their study, Kern et al. [20] aim to enhance ETA predictions generated by the Federal Aviation Administration (FAA)’s Enhanced Traffic Management System (ETMS). They select a set of baseline features and gradually increase complexity by adding new ones to check if they interfere with each other. The features are selected in 4 phases; 1) flight data, 2) flight and weather data, 3) flight and air traffic data, and 4) flight, weather, and air traffic data. Next, they apply Random Forest (RF) to generate the model. According to their results, their model predicts ETA 78.8% more accurately than the FAA’s ETMS. Although their model’s performance is applaudable, they use weather and air traffic data only for arrival airports during their feature collection, missing all the data along the routes between departure and arrival airports.

In his paper, Takacs [40] presents a solution to the General Electric’s Flight Quest contest to make flights more efficient by improving the accuracy of arrival estimates. The contestant presents his prediction approach to runway and gate arrival times of en route U.S. domestic flights based on flight history, weather, air traffic control, and other data available at a given time. His approach follows 6 consecutive stages of ridge regression and gradient boosting machines, trained on a total of 56 features. The model structure is a result of a heuristic, hand-run optimization process. The workflow iteratively defines features and tries to incorporate the modeling to decrease the RMSE as much as possible. However, due to time constraints imposed by the contest, the author decided to keep the model simple by using only 56 features.
Table 1: A set of major air routes in Spain.

<table>
<thead>
<tr>
<th>DepartureAirport</th>
<th>ArrivalAirport</th>
</tr>
</thead>
<tbody>
<tr>
<td>Barcelona–El Prat Airport (LEBL)</td>
<td>A Coruña Airport (LECO)</td>
</tr>
<tr>
<td></td>
<td>Málaga Airport (LEMG)</td>
</tr>
<tr>
<td></td>
<td>Vigo–Peinador Airport (LEVX)</td>
</tr>
<tr>
<td></td>
<td>Seville Airport (LEZL)</td>
</tr>
<tr>
<td>Adolfo Suárez Madrid-Barajas Airport (LEMD)</td>
<td>Almeria Airport (LEAM)</td>
</tr>
<tr>
<td></td>
<td>A Coruña Airport (LECO)</td>
</tr>
<tr>
<td></td>
<td>Jerez Airport (LEJR)</td>
</tr>
<tr>
<td></td>
<td>Menorca Airport (LEMH)</td>
</tr>
<tr>
<td></td>
<td>Palma de Mallorca Airport (LEPA)</td>
</tr>
<tr>
<td></td>
<td>Vigo–Peinador Airport (LEVX)</td>
</tr>
</tbody>
</table>

Glina et al. [15] use Quantile Regression Forests (QRF), which is an extension of RF as part of ensemble of Classification and Regression Tress (CART), to generate point predictions and pertinent conditional probability distributions for the ETA of individual flights. They validate their model on data from the Dallas/Fort Worth International Airport, obtaining mean absolute errors (MAE) of less than 60 seconds for their estimates of time-to-wheels-on for flights with a distance to the airport less than or equal to 20 nautical miles. Unfortunately, their data pertains to a single airport and it spans the period of only 5 days.

To summarize, our approach is distinguished from past efforts with the following respects: 1) Our system predicts ETA for runway arrival times. 2) Our system predicts ETA before departure, when the flight trajectory is still unknown. In fact, our prediction system doesn’t make use of costly flight plans, offering a more cost-effective solution. Hence, our approach addresses the ETA prediction problem strategically over time horizons of several hours. 3) For our model, we collect and use a richer set of features not only for the arrival airport but also for the airspace along the potential route. 4) We validate our model using an extensive set of real trajectory, weather, and airport data over the period of 11 months.

3 DATA DESCRIPTION

This section introduces the datasets used for ETA prediction.

3.1 Trajectory Data

Trajectory data plays a large role in our study. Trajectories have been the subject of much work in the spatial domain with an emphasis on cars along roads [37]. The focus has been on their generation (e.g., [38]), queries (e.g., [28, 30, 33, 35, 36]), and matching (e.g., [18, 27, 34]). This data is collected continuously and is quite voluminous. Instead, our focus here is on the flight domain.

Due to the fact that there exists no system that continuously records and stores exact positions of an aircraft’s original trajectory [4], only a discrete set of sample data are recorded and stored which presumably represent a close approximation of the original trajectory. We call this a raw trajectory. The raw trajectory data is provided by Spanish ANSP, ENAIRE, using radar surveillance feed with a 5 seconds update rate. The raw data is wrangled as part of the Data-driven AirCraft Trajectory prediction research (DART) project under the SESAR Joint Undertaking Work Programme [39].

The data contains all commercial domestic flights for Spain, a total of 119,563 raw trajectories and 80,784,192 raw trajectory points for the period of January through November 2016. The fields of the raw trajectory data are as follows: Flight No, Departure Airport, Arrival Airport, Date, Time, Aircraft Speed in X, Y, Z directions, and position information (Latitude, Longitude, Altitude). Note that, as a preprocessing step, we downsample raw trajectory data from the original resolution of 5 seconds to 60 seconds and align them to our 3D reference grid [1]. This is to build a set of features for each trajectory point, i.e. weather observations along the aligned trajectories. Table 1 lists a set of major flight routes departing from two major cities in Spain and Figure 2 provides their visual representation on a map.

3.2 Meteorology Data

The meteorology data is obtained from the National Oceanic and Atmospheric Administration’s (NOAA) Global Forecasting System (GFS) [26]. The original data has 28-km spatial and 6-hour temporal resolution and it contains over 40 weather parameters including Atmospheric, Cloud and Ground attributes for each grid point as part of its 3D weather model. Hence, for this study’s geographic volume of interest for the time period of January through November 2016, over 80TB of weather data is collected.

3.3 Airport Data

The airport traffic data is provided by EUROCONTROL as part of the Data-driven AirCrafTrajectory prediction research (DART) project under the SESAR Joint Undertaking Work Programme [39]. The data contains airport traffic data for Spain, a total of 1,252,571 sets of records, where each record is composed of Date, Flight No, Departure Airport, Arrival Airport, Aircraft type, Actual Departure Time, Actual Arrival Time, Scheduled Departure Time and Scheduled Arrival Time.

3.4 Airspace Data

The airspace sector data is computed using raw trajectory data along with airspace sector volumes. The data contains aircraft counts within 15-minute bins for each sector in Spain for a period of 11 months. The records include Sector Name, Date, Time of Sector Entry and Time of Sector Exit with counts of aircraft.
In this section, we select and construct a proper set of features for ETA prediction via feature engineering. Feature engineering refers to the process of using domain knowledge of data to create features that best represent the underlying problem of the predictive models, resulting in improved model accuracy on unseen data. We adopt high-dimensional features to build models so they possess the ability to predict ETA more accurately. Features extracted from each domain are listed in Table 2.

4.1 Basic Features

Basic features are extracted from each individual domain. We exploit Airline, Flight No and Aircraft Type as the flight features. The first three letters of Flight No identify the Airline. Intuitively, each of these basic flight features or combination of them exhibit a distinct pattern, potentially yielding a different ETA. For instance, as Aircraft Type, Bombardier Canadair Regional Jet (CRJ) has different performance parameters than that of Boeing 737-800’s (B738), likely to generate a different ETA.

We use Latitude, Longitude, Altitude, and Sector as the spatial features. Aircraft trajectories are nothing more than a set of joint 3D coordinates, where each coordinate is associated with a time stamp. Trajectories are usually of different length, formed by a different set of 3D coordinates. Hence, different spatial features may cause various patterns and biases in ETA prediction.

We adopt Date, Time, and Time Bin as the temporal features. Note that the temporal features listed in Table 2 are kept at a high level for the sake of saving space. Time bins are used to generate histograms. Note that, these features can refer to a flight departure or arrival or sector crossing as well as observation of a weather parameter. Essentially, ETA is an accumulated set of time intervals along the trajectory. Hence, a trajectory of 60 segments, where each segment is one-minute long, translates to an ETA of departure time plus 60 minutes.
Warping (DTW) we use 300 meteorological features for a flight that is 60-minutes with a machine learning approach utilized particular features such as weather observations and traffic data for the arrival airport only, disregarding features along the potential trajectory. This is partly due to fact that the exact trajectory is unknown before departure. However, excluding these key features along the potential trajectory greatly degrades the accuracy. Now, a critical question arises: How can we obtain a set of features along the potential trajectory when the trajectory is still unknown? Our approach to this problem is as follows: As a preprocessing step, we align raw historical trajectories to the 3D reference grid. The process generates a time series of aligned historical trajectories. Figure 3 illustrates a set of 3D grid points historically traversed by a single flight between a depart and arrival airport. Using spatially and temporally closest set of data points from the weather model, we perform interpolation and extract the pertinent parameters for each aligned trajectory point. Once weather observations have been extracted for each grid point at a particular time instance, we omit spatial information. Next, we perform Time Series Clustering with Dynamic Time Warping (DTW) on weather observations of variable length. The process aggregates the weather observations and generates a single set of representative features for each time instance. The result is a set of features in the form of time series. For instance, we use 300 meteorological features for a flight that is 60-minutes long. This is due to fact that each trajectory point is recorded once a minute and at each trajectory point 5 meteorological features are adopted along the potential route. The same applies to the sector congestion rates. For the same flight of 60 minutes, we use 4 time bins, resulting in 4 sector density features along the flight route.

Airport features consist of Arrival Airport, Departure Airport, Airport Arrival Count and Arrival Departure Count, in addition to Airport Congestion Rate, which is computed using the actual and scheduled arrival and departure counts. Each of these features are likely to have an impact on flight’s ETA, as the higher the airport congestion rate, the higher the likelihood of delay on arrival time.

We exploit Sector Aircraft Count and Sector Congestion Rate as airspace features. Similar to the impact by the airport congestion rate, the higher the airspace congestion rate along the planned route, the higher the likelihood of trajectory diversion causing a delay on arrival time.

### 4.2 Combinational Features

We combine various features from two or more domains to build combinational features. Combining flight and spatial features produces Flight-spatial combinational feature. Intrinsically, each airline’s flight exhibits a distinct trajectory pattern, yielding a different ETA. Figure 4a. shows Ryan Air (RYR) versus Vueling Airline’s (VLG) flights between the LEBL and LEZL airports. Red trajectories representing the RYR flight dominates the northern part, while white trajectories representing the VLG flight dominates the southern part of the routes.

Combining airport and temporal features to generate the Airport-temporal combinational feature. This enables us to gain insight on the distribution of aircraft counts on various airports at various time bins during the day, potentially impacting the ETA prediction. Figure 4c. captures the distribution of normalized arrival and departure counts within 15-minute time bins for the LEZL airport. The airport appears to have more traffic at time bins around 8:00.

To examine combination of various features and derive insights on trends, find anomalies and perform strategic planning, a novel interactive visualization tool, NormSTAD can be used.

### 4.3 Airport Congestion Rate

Given the actual (historical) arrivals and departures along with scheduled arrivals and departures for a particular airport, we want to compute the congestion rate for the 15-minute time bin, the flight of interest is scheduled to arrive. First, we create a set of 15-minute time bins throughout the day. Next, we compute the average counts for the actual arrival and departures for each of these 15-minute time bins. Given the scheduled arrival time of the flight of interest, we determine which time bin it falls into. For that particular time bin, we compute the airport congestion rate as presented in Algorithm 1. Figure 4d. illustrates congestion rates in each 15-minute bins at the LEZL airport for the flight VLG22XV, indicating a potential delay in late March due to a high congestion rate at the LEZL airport.

**Figure 4: Distribution of various features.**

**Algorithm 1: Airport Congestion Rate**

**Result:** Airport congestion rate

**Input:** Counts of scheduled arrivals $ctSA$, counts of scheduled departures $ctSD$, average count of actual arrivals $avgCtAA$, and average count of actual departures $avgCtAD$ of a flight with a flight number $fno$ and timestamp $ts$ for a particular airport.

**Output:** Congestion rate $cr$ of a particular airport, given the time stamp $ts$ and a flight number $fno$ of an arriving flight $fl$.

1. $TB \leftarrow \{tb_1, tb_2, ..., tb_k\}$
2. $3dp \leftarrow lat, lon, alt$
3. $fl \leftarrow fno, 3dp, ts$
4. foreach $tb \in TB$ do
5. if $fl.ts \subset tb$ then
7.  end
8. end

4.4 Sector Congestion Rate

Given the actual (historical) flights’ sector crossings, along with scheduled flights’ sector crossings for a particular sector, we want to compute the sector congestion rate for each sector, for each 15-minute time bin, the flight of interest is scheduled to cross. Due to the fact that airspace sectors are nothing more than a set of extruded polygons, we use a Point-In-Polygon algorithm to compute the counts of aircraft in each sector. To achieve this, we first create a set of 15-minute time bins throughout the day. Next, we compute the average counts for the actual flights’ sector crossings for each of these 15-minute time bins. Given the scheduled crossing time of the flight of interest, we determine which time bin it falls into, and which sector it likely crosses. Due to the fact that the flight’s trajectory is unknown at the time of ETA prediction, we consider all sectors the flight of interest is likely to cross based on actual sector crossings. Next, for that particular time bin, we compute the airport acceptance rate as presented in Algorithm 2.

5 PROBLEM STATEMENT AND MODEL REVIEWS

Given a set of historical flights with their attributes such as flight number, trajectory, departure and arrival airport, actual departure and arrival date and time, airline name, aircraft type, and associated weather and air traffic parameters, we aim at learning a model that predicts the current flight’s ETA before it departs. Note that ETA can refer to both runway and gate arrival times. Our system predicts runway arrival times i.e. times between aircraft wheels-off and wheels-on, as we exclude airport surface data when we build our model. To address this problem, we build a number of models, and rank them based upon their performance.
Weak learner is thereby forced to concentrate on the examples that may be of any functional form as long as 

\[
L_1 = \frac{|y_i(p(x_i) - y_i|}{D} \quad \text{(linear)}
\]

\[
L_2 = \frac{|y_i(p(x_i) - y_i|^2}{D^2} \quad \text{(squarelaw)}
\]

\[
L_3 = 1 - \exp \left(-\frac{|y_i(p(x_i) - y_i|}{D} \right) \quad \text{(exponential)}
\]

(5) Calculate an average loss: 

\[
L = \sum_{i=1}^{N_t} L_i p_i
\]

(6) Form \( \beta = \frac{L}{1-L} \) \( \beta \) is a measure of confidence in the prediction.

(7) Update the weights: \( w_i \rightarrow w_i \beta_l(1-L_i) \). The smaller the loss, the more weight is reduced making the probability smaller that this pattern will be selected as a member of the training set for the next machine in ensemble.

(8) For a particular input \( x_i \), each of the \( T \) machines makes a prediction. Obtain the cumulative prediction \( h_f \) using the T predictors:

\[
h_f = \inf \left\{ y \in Y : \sum_{t, k_j \leq y} \log(1/\beta_i) \geq \frac{1}{2} \sum_l \log(1/\beta_l) \right\}
\]

This is the weighted median. Equivalently, each machine \( h_f \) has a prediction \( y_i^{(1)} \) on the \( i \)th pattern and associated \( \beta_i \). For pattern \( i \) the predictions are relabeled such that for pattern \( i \) we have:

\[
y_i^{(1)} < y_i^{(2)} < \ldots < y_i^{(T)}
\]

Next, we sum the \( \log(1/\beta_i) \) until we reach the smallest \( t \) so that the inequality is satisfied. The prediction from that machine \( t \) we take as the ensemble prediction. If the \( \beta_i \) were all equal, this would be the median.

Intuitively, the effect of varying the weight \( w_i \) to give more emphasis to difficult examples means that each subsequent machine has a disproportionately harder set of examples to train on. Thus, the average loss tends to increase as we iterate through the algorithm and ultimately the bound on \( L \) is not satisfied and the algorithm terminates [10, 12]

### 5.2 Gradient Boosting

Gradient Boosting [13, 14] produces a prediction model in the form of an ensemble of weak prediction models. It builds additive regression models by sequentially fitting a simple parameterized function to current pseudo residuals by least squares at each iteration. The pseudo residuals are the gradient of the loss functional being minimized, with respect to the model values at each training data point, evaluated at the current step.

Formally, in the function estimation problem one has a system consisting of a random output variable \( y \) and a set of random input variables \( x = \{x_1, \ldots, x_n\} \). Given a training sample \( \{y_i, x_i\}_{i=1}^{N_t} \) of known \((y, x)\) values, the goal is to find a function \( F(x) \) that maps \( x \) to \( y \), such that over the joint distribution of all \((y, x)\) values, the expected value of some specified loss function \( L(y, F(x)) \) is minimized:

\[
F^*(x) = \arg\min_{F(x)} E_{y,x} L(y, F(x))
\]

Boosting approximates \( F^*(x) \) by an additive expansion:

\[
F(x) = \sum_{m=0}^{M} \beta_m h(x; a_m)
\]
Table 3: A list of prediction models used in this study.

<table>
<thead>
<tr>
<th>Method</th>
<th>Algorithm</th>
</tr>
</thead>
<tbody>
<tr>
<td>Linear</td>
<td>Linear Regression (LR)</td>
</tr>
<tr>
<td></td>
<td>Lasso Regression (LASSO)</td>
</tr>
<tr>
<td></td>
<td>Elastic Net Regression (EN)</td>
</tr>
<tr>
<td>Non-linear</td>
<td>Classification and Regression Trees</td>
</tr>
<tr>
<td></td>
<td>Support Vector Regression (SVR)</td>
</tr>
<tr>
<td></td>
<td>k-Nearest Neighbors (KNN)</td>
</tr>
<tr>
<td>Ensemble</td>
<td>Adaptive Boosting (AdaBoost)</td>
</tr>
<tr>
<td></td>
<td>Gradient Boosting (GBM)</td>
</tr>
<tr>
<td></td>
<td>Random Forest Regression (RF)</td>
</tr>
<tr>
<td></td>
<td>Extra Trees Regression (ET)</td>
</tr>
<tr>
<td>Recurrent Neural Network</td>
<td>Long Short-Term Memory (LSTM)</td>
</tr>
</tbody>
</table>

where the functions $h(x; a)$ which are base learners are usually chosen to be simple functions of $x$ with parameters $a = \{a_1, a_2, ..., a_k\}$. The expansion coefficients $\beta_m^M$ and the parameters $a_m^M$ are jointly fit to the training data in a forward stage-wise manner. One starts with an initial guess $F_0(x)$, and then for $m = 1, 2, ..., M$:

$$\beta_m^M, a_m^M = \arg\min_{\beta, a} \sum_{i=1}^{N} L(y_i, F_{m-1}(x_i) + \beta h(x_i; a))$$

and

$$F_m(x) = F_{m-1}(x) + \beta_m h(x; a_m^M)$$

Gradient boosting approximately solves $(\beta_m, a_m)$ for arbitrary loss functions $L(y, \hat{F}(x))$ with a two step procedure. First, the function $h(x; a)$ is fit by least-squares:

$$a_m = \arg\min_{a, \rho} \sum_{i=1}^{N} [\hat{y}_{im} - \rho h(x_i; a)]^2$$

to the current pseudo residuals:

$$\hat{y}_{im} = \left(\frac{\partial L(y_i, F(x_i))}{\partial F(x_i)}\right)_{F(x)=F_{m-1}(x)}$$

Then, given $h(x; a_m)$, the optimal value of the coefficient $\beta_m$ is determined:

$$\beta_m = \arg\min_{\beta} \sum_{i=1}^{N} L(y_i, F_{m-1}(x_i) + \beta h(x_i; a_m))$$

This strategy replaces a potentially difficult function optimization problem by one based on a least-squares, followed by a single parameter optimization based on a general loss criterion $L$.

6 EXPERIMENTAL STUDY

This section presents the evaluations of our ETA prediction system.

6.1 Setup

We evaluate the performance of our prediction system on 10 major flight routes in Spain, presented in Table 1. We chronologically order each dataset for each route, normalize and standardize, and use the first 80% for modeling and the remaining 20% for validation. Table 3 lists the prediction models used in this study. We use 10-fold cross-validation and evaluate the algorithms using Root Mean Squared Error (RMSE) metric. We create a baseline of performance on this problem and spot-check a number of algorithms to address the regression problem. The first round of algorithms we use are linear and non-linear regression methods, in addition to historical average (HA), which simply averages all historical flight times for the same routes for the same time period. The algorithms all use default tuning parameters. The second round of algorithms we use are ensemble methods comprised of two boosting (AB and GBM), and two bagging (RF and ET) methods. These algorithms also use default tuning parameters. The final algorithm we use is the Long Short-Term Memory (LSTM), which is a type of RNN. Given the flight times for the previous 10 most recent subsequent days, we compute the time for the next day’s flight, provided that all flights share the same defining features such as flight number, departure and arrival airports, aircraft type, etc. In order to perform LSTM on our dataset, we first transform our multivariate time series data into a supervised learning problem. Then, we define the LSTM with 50 neurons in the first hidden layer and 1 neuron in the output layer for predicting the flight time. The input shape is one time step with over 300 features (the number of features depends on the average flight time). The model is fit for 50 training epochs with a batch size of 72.

Our experiments are conducted on a computer with Intel Core i7-6820HQ CPU @ 2.70GHz and 16GB memory, running on Linux Ubuntu 16.04.1 64-bit Operating System. All the algorithms in our system are implemented in Python 3.6.4.

6.2 Results

Table 4 shows the RMSE value for each algorithm. The best score for each route is highlighted. In a quick glance, it seems clear that HA and linear models (LR, LASSO, EN) perform poorly on all routes. KNN and SVR seem to achieve the best scores among the linear and non-linear methods. In fact, KNN generates 8, SVR generates 2 best scores out of 10 routes among the linear and non-linear methods. The default value for the number of neighbors in KNN is set to 7. We obtain slightly better results when we use a grid search to try different numbers of neighbors. Unlike the previous work [15, 20] which emphasize the accuracy of bagging methods (RF and its extensions), our results indicate that boosting methods (AB and GBM) dominate the bagging methods (RF and ET) on all routes. LSTM yields relatively unstable scores, generating only one best score on all 10 routes. This may not be a surprise as LSTM is better suited for sequence prediction than autoregression.

Overall, the boosting methods dominate the entire list, generating the best scores among all including the linear, non-linear, ensemble and RNN methods. In fact, the scores obtained by boosting methods can be further improved with tuning. The default number of boosting stages to perform (n estimators) is originally set to 100. We define a parameter grid n estimators values from 50 to 400 in increments of 50. Each setting is evaluated using 10-fold cross-validation. By parameter tuning, we reach even better scores.

We complete our comparative study among the models listed in Table 3 and select the best score for each route. Next, we compare our final results with the ETA values, EUROCONTROL uses [39].
Table 4: The RMSE values of all algorithms.

<table>
<thead>
<tr>
<th>Model</th>
<th>LEBL</th>
<th>LECO</th>
<th>LEMG</th>
<th>LEVM</th>
<th>LEVX</th>
<th>LEBL</th>
<th>LEZL</th>
<th>LEAM</th>
<th>LEMD</th>
<th>LECO</th>
<th>LEMD</th>
<th>LEJR</th>
<th>LEMD</th>
<th>LEHG</th>
<th>LEMD</th>
<th>LEP</th>
<th>LEMD</th>
</tr>
</thead>
<tbody>
<tr>
<td>LASSO</td>
<td>5.423176</td>
<td>5.328147</td>
<td>5.466031</td>
<td>5.620595</td>
<td>2.979159</td>
<td>3.939358</td>
<td>3.061945</td>
<td>3.618408</td>
<td>3.830357</td>
<td>5.167527</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>EN</td>
<td>5.129900</td>
<td>5.328147</td>
<td>5.410600</td>
<td>4.988698</td>
<td>2.979159</td>
<td>3.947221</td>
<td>3.078638</td>
<td>3.618059</td>
<td>3.759207</td>
<td>5.562907</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Figure 5 illustrates RMSE values in minutes for each route between our prediction versus EUROCONTROL’s prediction. Note that Figure 5 presents both results at two different scales. Figure 5a is a closer look at the box plots, where the median values are visible. However, the full extent of the boxplots are missing in Figure 5a due to outliers. Hence, we provide Figure 5b, where the full extent of the boxplots including the outliers are visible. From the results, we make the following observations: i) Our prediction yields better median scores on eight routes, while the EUROCONTROL’s ETA shows better median scores on two routes (LEBL-LEVX and LEBL LEZL). ii) The standard deviation values in EUROCONTROL’s ETAs are much larger, resulting in larger windows of predictability at arrival times. This may have a significant adverse impact on airlines’ ground resource management as well as connecting flights’ scheduling. iii) Boxplots representing EUROCONTROL’s ETAs in Figure 5b, show extreme outliers. In summary, our prediction system offers more accurate ETA prediction with far smaller standard deviation than those of EUROCONTROL.

To evaluate the effectiveness of our features, we rank them based on their relative importance. The top 10 features are presented in Table 5. Evidently, the arrival airport is of critical importance on accurate ETA prediction. Aside from that, meteorological features in the form of a time series along the potential trajectory may have a significant impact during regression. This insight validates our approach that meteorological and airspace sector features along the potential trajectories should be taken into account to achieve accurate ETA prediction.

Likewise, air traffic flow management, airport runway and gate assignment, and ground support equipment usage optimization are all processes that can benefit from a more accurate ETA prediction for commercial flights.

Some future work could involve adding a spatial browsing capability [8, 11, 32] for the trajectories as well as incorporating our methods in distributed spatial environment [41].

Table 5: Top 10 features ranked by their relative importance.

<table>
<thead>
<tr>
<th>Rank</th>
<th>Feature</th>
<th>Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Arrival airport</td>
<td>1.0</td>
</tr>
<tr>
<td>2</td>
<td>Atmospheric pressure</td>
<td>0.67854</td>
</tr>
<tr>
<td>3</td>
<td>Atmospheric wind speed</td>
<td>0.66231</td>
</tr>
<tr>
<td>4</td>
<td>Atmospheric wind direction</td>
<td>0.65224</td>
</tr>
<tr>
<td>5</td>
<td>Atmospheric humidity</td>
<td>0.63331</td>
</tr>
<tr>
<td>6</td>
<td>Atmospheric temperature</td>
<td>0.61314</td>
</tr>
<tr>
<td>7</td>
<td>Airport congestion rate</td>
<td>0.53212</td>
</tr>
<tr>
<td>8</td>
<td>Sector congestion rate</td>
<td>0.31153</td>
</tr>
<tr>
<td>9</td>
<td>Flight no</td>
<td>0.29192</td>
</tr>
<tr>
<td>10</td>
<td>Aircraft type</td>
<td>0.13221</td>
</tr>
</tbody>
</table>

8 ACKNOWLEDGEMENTS

This work was supported in part by the NSF under Grant IIS-13-20791. We are grateful to the Boeing Research & Technology Europe, a member of the project DART [20] team for providing the data used in this study. The project DART has received funding from the SESAR Joint Undertaking under Grant Agreement No. 699299 under the European Union’s Horizon 2020 Research and Innovation Programme.

REFERENCES

Figure 3: RMSE by our prediction vs EUROCONTROL’s on all flight routes.