pp 1245–1259. © The Author(s), 2021. Published by Cambridge University Press on behalf of Royal Aeronautical Society.
doi:10.1017/aer.2021.13

Prediction of aircraft estimated time of arrival using machine learning methods

O. Basturk
onur.basturk@adalet.gov.tr
Republic of Turkey Ministry of Justice
Directorate General for Information Technologies
Ankara
Turkey

C. Cetek
Eskisehir Technical University
Eskisehir
Turkey

ABSTRACT

In this study, prediction of aircraft Estimated Time of Arrival (ETA) is proposed using machine learning algorithms. Accurate prediction of ETA is important for management of delay and air traffic flow, runway assignment, gate assignment, collaborative decision making (CDM), coordination of ground personnel and equipment, and optimisation of arrival sequence etc. Machine learning is able to learn from experience and make predictions with weak assumptions or no assumptions at all. In the proposed approach, general flight information, trajectory data and weather data were obtained from different sources in various formats. Raw data were converted to tidy data and inserted into a relational database. To obtain the features for training the machine learning models, the data were explored, cleaned and transformed into convenient features. New features were also derived from the available data. Random forests and deep neural networks were used to train the machine learning models. Both models can predict the ETA with a mean absolute error (MAE) less than 6min after departure, and less than 3min after terminal manoeuvring area (TMA) entrance. Additionally, a web application was developed to dynamically predict the ETA using proposed models.

Keywords: Aircraft arrival time prediction; ETA prediction; Machine learning; Random forests; Deep neural networks; Deep learning

NOMENCLATURE

ADS–B	automatic dependent surveillance–broadcast
AMAN	arrival manager
ANSP	air navigation service provider

Received 16 November 2020; revised 1 February 2021; accepted 15 February 2021.

https://doi.org/10.1017/aer.2021.13 Published online by Cambridge University Press

ATD	actual time of departure
ATM	air traffic management
CART	classification and regression trees
CDM	collaborative decision making
DDR2	EUROCONTROL Demand Data Repository
DNN	deep neural network
ETA	estimated time of arrival
ETMS	enhanced traffic management system
FAA	Federal Aviation Administration
IATA	International Air Transport Association
ICAO	International Civil Aviation Organization
ISG	Istanbul Sabiha Gokcen International Airport
LTBA	Istanbul Atatürk International Airport
MAE	mean absolute error
MCNN	multi-cells neural network
METAR	meteorological aerodrome report
MOJO	model object optimised
MSE	mean squared error
NN	neural networks
OOB	out-of-bag
QRF	quantile regression forests
RF	random forests
RMSE	root mean squared error
RNN	recurrent neural network
SESAR	Single European Sky ATM Research
STA	scheduled time of arrival
STD	scheduled time of departure
TMA	terminal manoeuvring area

1.0 INTRODUCTION

Over the next 20 years, a critical growth in commercial air traffic is expected globally[1,2]. Novel concepts of operations are being implemented in the context of Single European Sky Air Traffic Management (ATM) Research (SESAR) and NextGen to accommodate this growth. However, the capacity and the efficiency of the airspace are defined deterministically, and usually the external operational conditions that affect the actual flight profiles directly are not considered[3]. Flight delays mean extra cost for the airlines because of the increase in fuel consumption, emissions and additional flight hours[4] leading to passenger dissatisfaction and market loss for airlines. They also have effects on the whole network by impacting the following flights because of misconnections, reduced turnaround times and crew roster changes.

Accurate prediction of Estimated Time of Arrival (ETA) is a key component of delay management by trading accelerated fuel burn against the cost of the delay. It is also very important

https://doi.org/10.1017/aer.2021.13 Published online by Cambridge University Press

for Collaborative Decision Making (CDM), where costs are kept under control with improvements in four key ATM areas: safety, capacity, efficiency and environmental impact. Accurate ETA prediction makes it easier for the airlines to co-ordinate the ground operations, reducing the waste of time and energy. Air traffic flow management, airport runway and gate assignment are some of the processes that could be handled much more easily with a better ETA prediction. Accurate ETA prediction will also help to improve the performance of decision support tools used for sequence optimisation such as arrival managers (AMAN) and will provide air traffic controllers valuable situational awareness[5].

Traditional methods for predicting ETA usually rely on aircraft performance models and parametric or physics-based trajectory models[6–8]. First the flight trajectory is estimated, including the flight path, altitude and speed profiles, and then the time required to match the trajectory is calculated. However, these models do not take into account the external operational conditions that affect the actual flight profile directly, such as the weather conditions, airspace sector densities or airport congestion. As its name implies, machine learning can learn even very complex functions from experience and real data. Therefore, machine learning methods will consider both the performance of the aircraft and the external operational circumstances that affect the flight. Compared with traditional parametric or physics-based trajectory models, machine learning models are built with fewer or even no assumptions.

In recent years, machine learning methods have been successfully applied to aircraft ETA prediction. Kern et al. introduced a method for improving the aircraft ETA predictions using machine learning methods, considering general information about the flight, weather and air traffic data. The main goal was to improve ETA predictions generated by the Federal Aviation Administration's (FAA) Enhanced Traffic Management System (ETMS), which depends on aircraft performance models and parametric kinematics-based trajectories[4]. The random forests (RF) method[9] was used to train the model which improved the ETA prediction provided by the ETMS, with a 42.7% Mean Absolute Error (MAE) reduction. Takacs proposed a six-stage model using successive ridge regressions and gradient boosting machines in a data science competition called GE Flight Quest, where the goal was to improve the flight efficiency by improving the estimation accuracy of arrival times[10]. Takacs' solution placed third in the competition. Glina et al. presented a novel approach for the estimation of landing times using a tree-based ensemble method called Quantile Regression Forests (QRF)[5]. The approach was tested for arrivals at Dallas/Fort Worth International Airport with 5-day data, and the ETA of aircrafts closer than 20nm to the airport were predicted with a MAE less than 60s. Ayhan et al. collected a rich feature set by including the weather data along the potential flight path and obtained more accurate ETAs than those of EUROCONTROL using regression models and Recurrent Neural Network (RNN)[3]. Wang et al. introduced a hybrid model for short-term trajectory prediction in Terminal Manoeuvring Area (TMA) by application of machine learning methods[11]. The model included two parts: pre-processing based on clustering and prediction based on Multi-Cells Neural Network (MCNN). The trajectories were clustered into several patterns, and then the MCNN model was used to determine the ETA prediction model per cluster. Using Beijing TMA trajectory data, they demonstrated that the proposed model was very effective and robust, both in trajectory clustering and short-term four-dimensional (4D) trajectory prediction.

In this study, ETA prediction of arrivals to Istanbul Atatürk International Airport (LTBA) is proposed using deep neural networks and RF methods. The proposed approach differs from the previous work in various ways. Flights with holding patterns and deviations from the standard arrival routes are not considered as outliers. Instead of using departure–arrival airport pairs as categorical variables, total flight distances calculated from the trajectories were used

https://doi.org/10.1017/aer.2021.13 Published online by Cambridge University Press

to train the model, which allows the same model to be used with different arrival airports having similar runway configurations. The machine learning models were trained in less than 3min using a quad-core computer with a 2.80GHz processor and 6GB memory. The models are able to predict dynamically the ETA for any flight in less than 5s. The prediction errors were decreased 64% after departure and 84% after TMA entrance with respect to scheduled time of arrival. Finally, a web application was developed to predict the ETA of any individual flights to LTBA dynamically using the generated machine learning models.

The proposed method can be used by Air Navigation Service Providers (ANSPs) or airlines for more accurate planning and resource management, which will have a positive impact on four main performance areas of ATM. It will also increase situational awareness of air traffic controllers and help them to sequence the arrival traffic using decision support tools.

The paper is organised as follows. In Section 2, data used for training the machine learning methods are described in detail. In Section 3, RF and deep learning methods are explained. In Section 4, the experimental results of machine learning models built to predict the ETA of aircraft landing at Istanbul Ataturk Airport are provided. Finally, conclusions and future work are presented in Section 5.

2.0 DATA DESCRIPTION

The raw data used for ETA prediction consist of general information about the flight, trajectory points and airport weather reports. The raw data were obtained in different formats (e.g. Automatic Dependent Surveillance-Broadcast (ADS-B), allft+, Meteorological Aerodrome Report (METAR)) from different data sources (e.g. EUROCONTROL Demand Data Repository (DDR2), Iowa State University). Then the data were exported to a relational database which was designed to be able to query and explore the data easily. A Java web application was developed to visualise the data (trajectories, TMA entrances etc.) on the map, which provides further insight into the possible problems and issues related with the data. The features to be used in training the machine learning models were determined depending on the available data, domain knowledge and previous studies. Then, a feature engineering process is employed to remove the missing, inconsistent (e.g. flights with incomplete trajectory points) or duplicate data and to derive new features from available data using data transformations.

2.1 Dataset

The main dataset used in this study consists of arrivals to Istanbul Atatürk International Airport (IST/LTBA) from April 2018 to August 2018. The dataset contains 63,460 commercial flights (both domestic and international), 18,869,049 trajectory points and 6,236 METAR reports for LTBA. Allft+ data for the same period were also obtained from DDR2 and were used to determine the time and co-ordinates of the TMA entrance for each aircraft. Table 1 presents a list of data fields related with the general flight information, flight trajectory and weather around airport.

2.2 Feature engineering

In machine learning projects, it is crucial to convert the raw data into a relational and tidy data format (e.g. a database) to be able to easily explore the relationship among data. Therefore, a relational database was designed to establish the relationship between the data obtained in different formats from different sources. Then, the data were inserted into the database from various data files. A Java web application was developed for data exploration and visualisation

https://doi.org/10.1017/aer.2021.13 Published online by Cambridge University Press

Table 1
Data fields

General flight information	Trajectory data	Airport weather reports
ICAO/IATA callsign	Time over point	Report time
Departure airport	Latitude	Wind direction
Arrival airport	Longitude	Wind speed
Airline	Altitude	Temperature
Aircraft type	Ground speed	Dew point
Scheduled time of departure	Heading	Pressure
Actual time of departure		Visibility
Scheduled time of arrival		
Actual time of arrival		

Table 2
Timestamp transformation

Timestamp	Day of year	Time, min
01 April 2018, 09:50	91	590
28 May 2018, 17:15	148	1,035

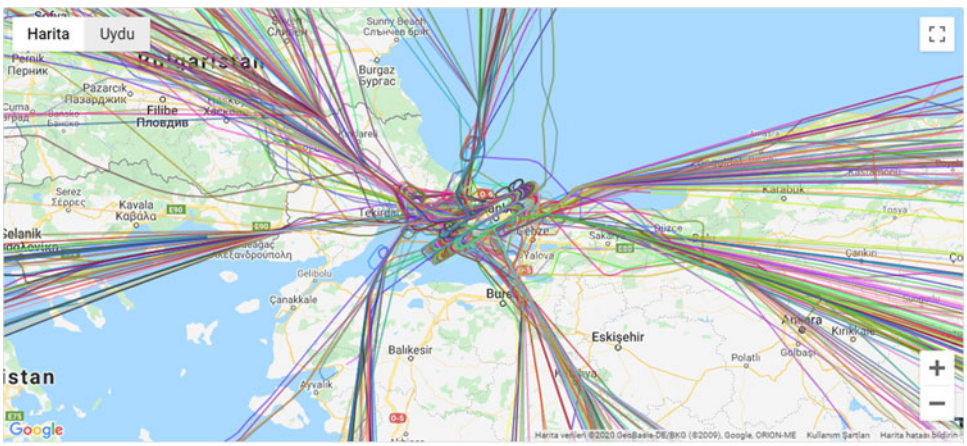


Figure 1. Trajectories of arrivals to LTBA on 22 July 2018.

purposes. Figure 1 shows the trajectories of arrivals to LTBA for one of the busiest days in the selected period. After a careful data exploration process, flights with missing feature values and duplicate data were removed. Data with inconsistencies (very close TMA and runway arrival times, incomplete trajectories, etc.) were also removed. After data exploration and cleaning, we had 51,902 individual flights in the dataset.

Timestamp data are not appropriate for training machine learning models. Therefore, the timestamp data such as scheduled and actual departure/arrival times were converted into two numerical features: ‘day of year’ and ‘minutes passed since midnight’. Table 2 presents

https://doi.org/10.1017/aer.2021.13 Published online by Cambridge University Press

Table 3
METAR conversion example

	LTBA 010050Z 16004KT CAVOK 11/08 Q1010 NOSIG
Report date	01 April 2018, 00:50
Wind direction	160
Wind speed	4
Temperature	11
Dew point	8
Pressure	1,010
Visibility	9,999

examples of this transformation. The aircraft types were grouped into three categories corresponding to International Civil Aviation Organization (ICAO) Wake Turbulence Categories: 'light', 'medium' and 'heavy'. More than 90% of the flights in the dataset were operated by national airlines. Therefore, airline categories were grouped into five categories: 'Turkish Airlines', 'Atlas Global', 'Onur Air', 'Pegasus Airlines' and 'Other'.

Four additional features were derived from the data: the arrival runway, aircraft type (e.g. ICAO wake turbulence category), total flight distance and the distance from TMA entrance to runway. The arrival runway was determined from the arithmetic average of the headings of final approach trajectory points. The total distance was calculated using the trajectory data. Therefore, the effect of the departure airport was not considered to be the same for all flights departing from the same airport. To calculate the distance from TMA entrance to runway, the co-ordinates of the TMA entrance were determined using allft+ data which contains the airspace entry/exit time and co-ordinates. METAR data were parsed from their original format. METARs closest to the actual arrival times of the corresponding flights were determined and matched with the flight data to reflect the actual weather conditions around the airport at the time of arrival. Table 3 presents an example of METAR conversion, and Table 4 presents the final set of features used for ETA prediction. Figure 2 shows the whole feature engineering process.

3.0 METHODOLOGY

RF and deep neural networks were used for ETA prediction. RF[9] is an ensemble learning method to build a forest of uncorrelated decision trees using a procedure such as Classification and Regression Trees (CART)[12], with random split selection[13] and bagging[14]. A decision tree consists of many decision nodes, beginning at the root node and ending at the leaf nodes (Fig. 3). Nodes are connected to sub-nodes with branches which are the possible outcomes of the decisions given at the parent nodes[15]. At each decision node m, a test function $f_m(x)$ is applied to a given input, and one of the branches is taken depending on the output[16]. This process is repeated until a leaf node is hit. The value of the leaf node is the output.

Each decision node uses only one of the input features (x_j) in a univariate tree. If x_j is discrete with n possible values, then the decision node will have n splits. The corresponding branch will be selected depending on the value of x_j. If x_j is numeric, then $f_m(x)$ will compare

https://doi.org/10.1017/aer.2021.13 Published online by Cambridge University Press

Table 4
Features for ETA prediction

Feature	Data type
Scheduled Time of Departure (STD)	Numerical
Actual Time of Departure (ATD)	Numerical
Scheduled Time of Arrival (STA)	Numerical
Total distance	Numerical (derived)
Distance from TMA entrance to runway	Numerical (derived)
TMA entrance time	Numerical
Airline	Categorical
Aircraft type	Categorical (derived)
Arrival runway	Categorical (derived)
Wind speed around the airport	Numerical
Wind direction around the airport	Numerical
Temperature around the airport	Numerical
Visibility around the airport	Numerical

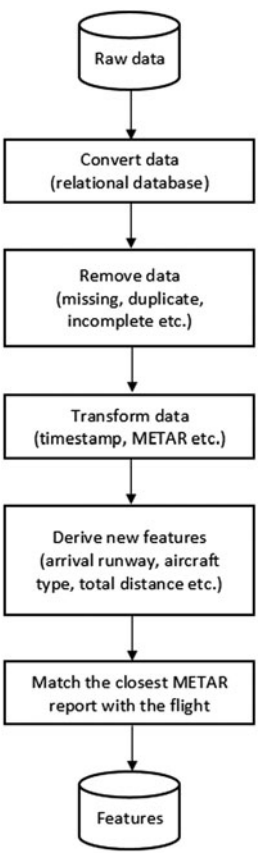


Figure 2. Feature engineering process.

https://doi.org/10.1017/aer.2021.13 Published online by Cambridge University Press

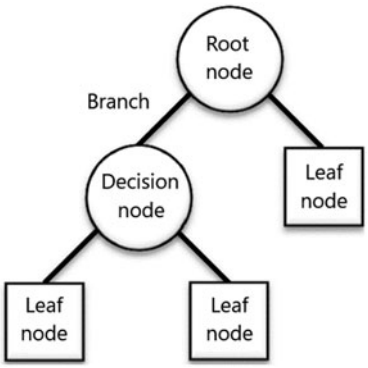


Figure 3. Decision tree.

x_j with a threshold value w_{m_0} and divide the input space into two regions. This is called a binary split.

$$L_m = \{x | x_j > w_{m_0}\} \quad \ldots (1)$$

$$R_m = \{x | x_j \leq w_{m_0}\} \quad \ldots (2)$$

Many different decision trees can be constructed for a given training set. Usually, the smallest tree is tried to be found greedily by looking for the best split at each step. In a classification tree, where the output of the decision tree is one of the p classes, the goodness of a split is determined by an impurity measure. In a two-class problem where $p^1 = p$ and $p^2 = 1 - p$, the impurity of a split can be measured using a non-negative function $\phi\,(p, 1-p)$. Some examples of this function are entropy, Gini index and misclassification error, which are respectively defined as:

$$\phi\,(p, 1-p) = -p\log_2 p - (1-p)\log_2\,(1-p) \quad \ldots (3)$$

$$\phi\,(p, 1-p) = 2p\,(1-p) \quad \ldots (4)$$

$$\phi\,(p, 1-p) = 1 - \max\,(p, 1-p) \quad \ldots (5)$$

In a regression tree, where the output of the decision tree is a numeric value, the goodness of a split is measured by the mean square error from the estimated value.

RF consists of many decision trees where the output of the forest is determined by majority voting or averaging in classification and regression problems, respectively (Fig. 4). RF can handle both continuous and categorical features, and they can run efficiently on large data sets with many features. Additionally, overfitting is prevented with the randomness and the use of decision trees as an ensemble. Each tree in the RF is grown using random samples of the training data. This is called bagging. The samples that are excluded in the training process are called 'Out-Of-Bag' (OOB) samples. They can be used to get unbiased prediction error estimates and produce feature importance estimates[4]. At each node, the split is selected randomly among the K best splits.

https://doi.org/10.1017/aer.2021.13 Published online by Cambridge University Press

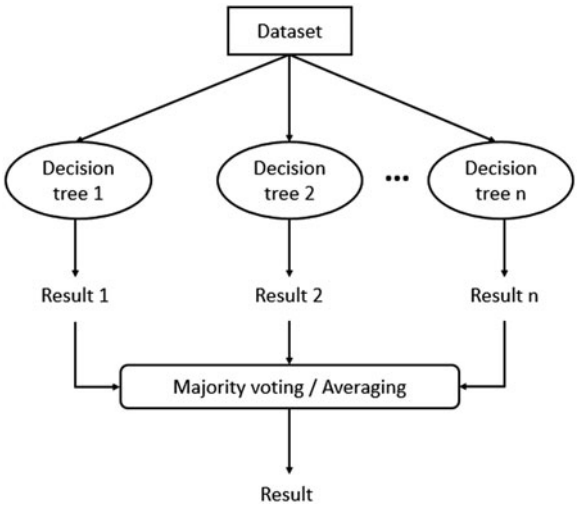


Figure 4. Random forests.

The characteristics of RF are as follows[9]:

i. RF is robust to outliers and noise.
ii. RF works faster than bagging or boosting.
iii. RF provides useful estimates of error, variable importance and correlation.
iv. RF is simple and comprehensible.

The term 'deep learning' was introduced by Dechter in 1986[17]. It was first mentioned in Neural Networks (NNs) by Aizenberg et al. in 2000[18]. Deep learning is the state-of-the art method in speech recognition, visual object recognition, object detection, etc. Deep learning models consist of multiple layers which reveal the complex relationships in large data sets and learn the representations of data using the backpropagation algorithm[19].

At each layer, deep learning methods transform the current representation into a more abstract representation. Very complex functions can even be learned with such transformations. Figure 5 shows the representation of a deep neural network with k input features, m hidden layers and n neurons (nodes) in each hidden layer. Each node in the deep neural network corresponds to a computational unit that takes the weighted sum of node inputs and passes the result to a non-linear activation function to calculate the output of that node (Fig. 6). Nodes that are not in the input or output layer are called hidden nodes. Deep neural networks (DNNs) are the neural networks with multiple hidden layers.

Here,

$$x = \begin{bmatrix} x_1 & x_2 & \dots & x_k \end{bmatrix}^T = \text{feature vector},$$

$$w = \begin{bmatrix} w_1 & w_2 & \dots & w_k \end{bmatrix}^T = \text{weight vector},$$

$b =$ bias vector,
$a =$ activation function

https://doi.org/10.1017/aer.2021.13 Published online by Cambridge University Press

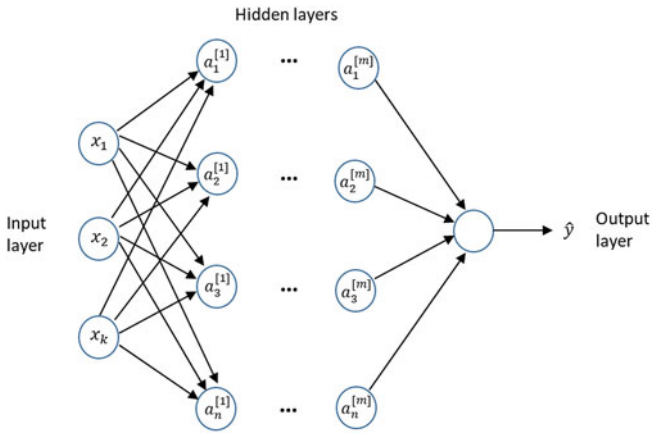


Figure 5. Neural network representation.

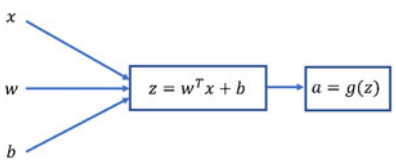


Figure 6. Structure of a single neuron.

Some of the non-linear activation functions are as follows:

$$f(z) = \max(0, z) \quad (ReLU) \qquad \ldots(6)$$

$$f(z) = \frac{e^{z} - e^{-z}}{e^{z} + e^{-z}} \quad (tanh) \qquad \ldots(7)$$

$$f(z) = \frac{1}{1 + e^{-z}} \quad (sigmoid) \qquad \ldots(8)$$

$$f(z) = z \quad (identity) \qquad \ldots(9)$$

Equations (10)–(16) define the procedure to calculate the predicted output of the deep neural network shown in Fig. 5 starting with the layer 1 through layer $\boldsymbol{m}$.

$$z^{[1]} = W^{[1]}x + b^{[1]} \qquad \ldots(10)$$

$$a^{[1]} = f\left(z^{[1]}\right) \qquad \ldots(11)$$

$$z^{[2]} = W^{[2]}a^{[1]} + b^{[2]} \qquad \ldots(12)$$

$$a^{[2]} = f\left(z^{[2]}\right) \qquad \ldots(13)$$

https://doi.org/10.1017/aer.2021.13 Published online by Cambridge University Press

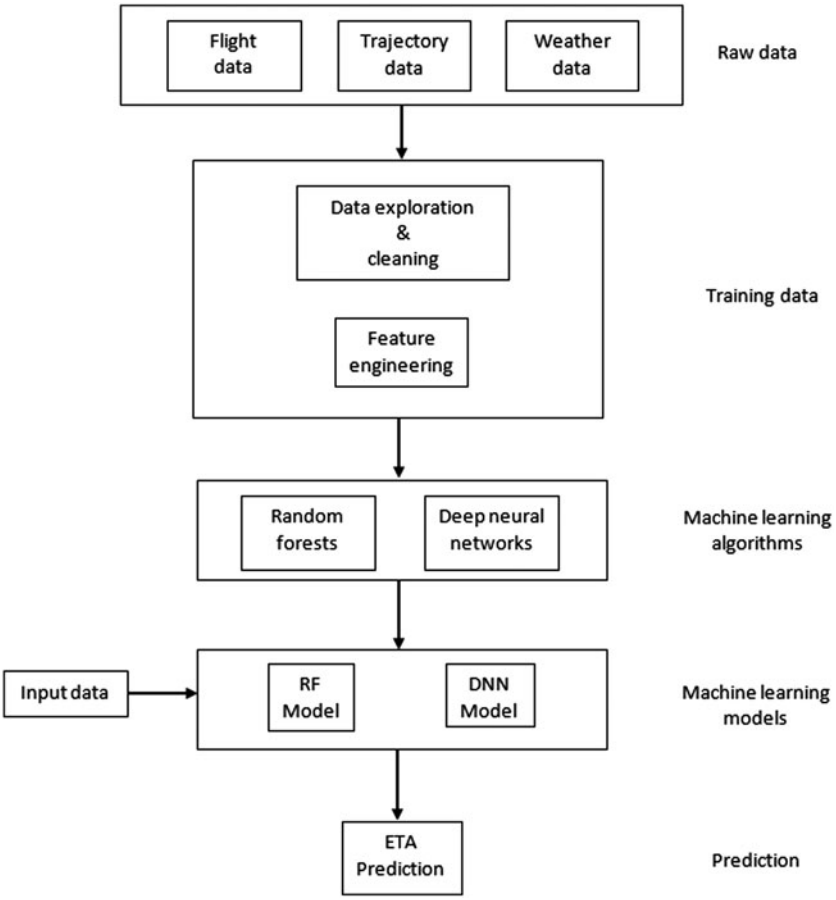


Figure 7. ETA prediction approach.

$$z^{[m]} = W^{[m]}a^{[m-1]} + b^{[m]} \quad \ldots (14)$$

$$a^{[m]} = f\left(z^{[m]}\right) \quad \ldots (15)$$

$$\hat{y} = f(a^{[m]}) \quad \ldots (16)$$

Here $\hat{y}$ is the output vector.

Simple stochastic gradient descent can be used to train multi-layer architectures. Gradients can be computed using the backpropagation algorithm. The backpropagation algorithm is actually a practical application of the chain rule for derivatives to calculate the parameters of the model (W, b) that minimise the prediction error.

Figure 7 shows the steps of the proposed ETA prediction approach. Raw flight, trajectory and weather data are filtered and formatted by data exploration and cleaning, and then training data were obtained through feature engineering processes. RF and DNN algorithms were used to train and test machine learning models. ReLU activation function given in Equation (6) and identity activation function given in Equation (9) were used in the hidden and output layers of

https://doi.org/10.1017/aer.2021.13 Published online by Cambridge University Press

Table 5
Results for ETA prediction after departure (LTBA)

	RF	DNN
MAE	5.68	5.91
MSE	66.40	68.14
RMSE	8.14	8.25

Table 6
Results for ETA prediction after TMA entry (LTBA)

	RF	DNN
MAE	2.60	2.66
MSE	13.35	12.36
RMSE	3.65	3.52

DNN, respectively. These models were then used in an application to predict the ETA using any input data.

MAE, Mean Squared Error (MSE) and Root Mean Squared Error (RMSE) metrics given in Equations (17)–(19) are used to evaluate the performance of ETA prediction models.

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|\widehat{y_i} - y_i| \quad \ldots(17)$$

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(\hat{y}_i - y_i)^2 \quad \ldots(18)$$

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\hat{y}_i - y_i)^2} \quad \ldots(19)$$

where n = number of observations, $\hat{y}_i = i-$ th predicted value and y_i = the i-th observed value of ETA.

4.0 EXPERIMENTAL RESULTS AND DISCUSSION

RF and deep neural networks were implemented in R programming language using h2o package, an open source, distributed and scalable machine learning platform[20]. As mentioned in Section 2.2, flight information data, trajectory data and METAR data for LTBA were used for training the machine learning models. The data were randomly split into training (80%) and test data (20%). The performances of machine learning models were evaluated using various parameter values. Parameters for RF were chosen as number of split features = 6 and the number of trees = 200. Parameters for deep learning were chosen as number of epochs = 100, number of hidden layers = 2 and number of neurons per hidden layer = 40. A quad-core computer with a 2.80GHz processor and 6GB memory was used for training and running the model. Training took less than 3min. Tables 5 and 6 show the evaluation metrics for ETA

https://doi.org/10.1017/aer.2021.13 Published online by Cambridge University Press

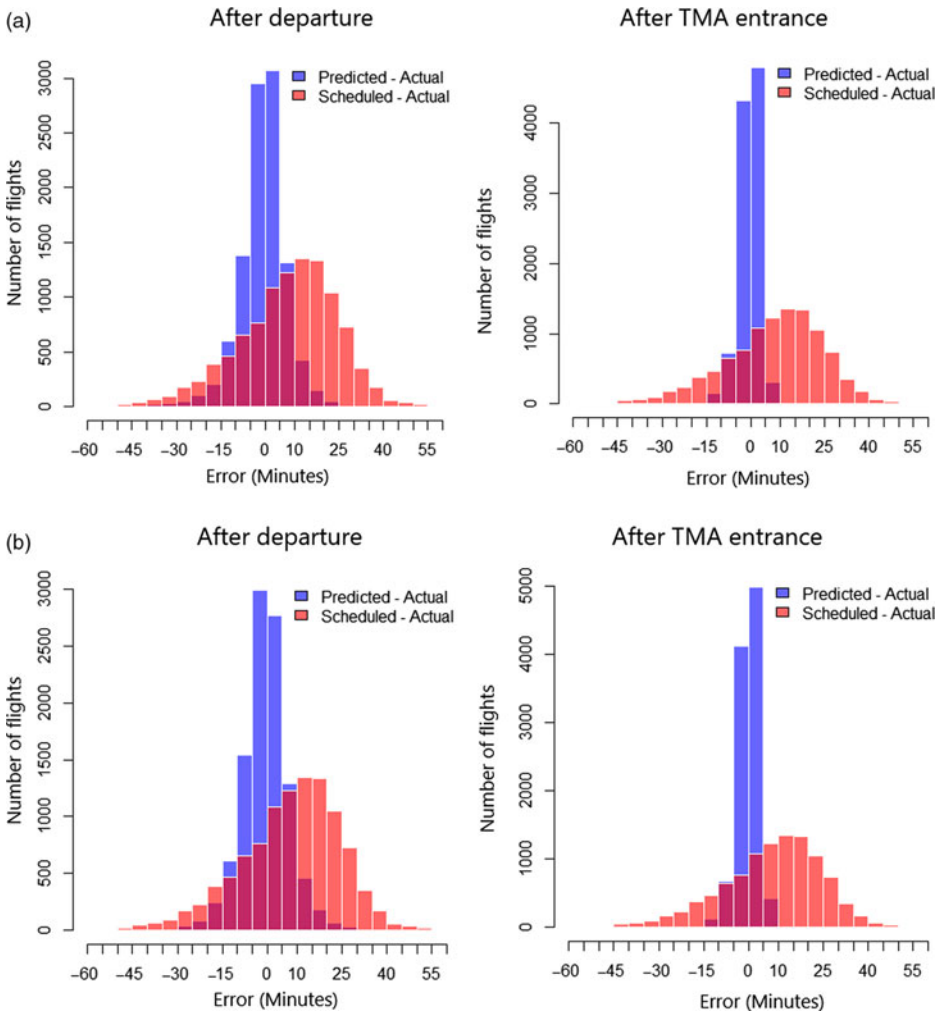


Figure 8. Error histograms after departure and after TMA entrance using (A) RF and (B) DNN.

prediction using RF and DNN after departure and TMA entry, respectively. ETAs after TMA entrance can be predicted with a MAE less than 3min, while ETAs after departure can be predicted with a MAE less than 6min. Machine learning models decreased the prediction errors 64% after departure and 84% after TMA entrance with respect to scheduled time of arrival. Figure 8 shows the error histograms for ETA prediction using RF and DNN. The bars entitled 'Scheduled-Actual' in the error histograms represent the error between the scheduled time of arrival and the actual time of arrival. The horizontal axis represents the error in minutes, while the vertical axis represents the number of flights. The bars entitled 'Predicted-Actual' represent the error between the time of arrival predicted by the machine learning models and the actual time of arrival. The error histograms clarify that predictions calculated by the machine learning models concentrate around a narrow vicinity of 0, where '0' means an exact match with the actual time of arrival. It is much clearer in the error histograms for the predictions calculated after TMA entrance, where it can be seen that more than 90% of the errors are between −5min and 5min. On the contrary, the errors between the scheduled and actual time

https://doi.org/10.1017/aer.2021.13 Published online by Cambridge University Press

Table 7
Results for ETA prediction after departure (ISG)

	RF	DNN
MAE	8.83	8.36
MSE	188.33	128.82
RMSE	13.72	11.35

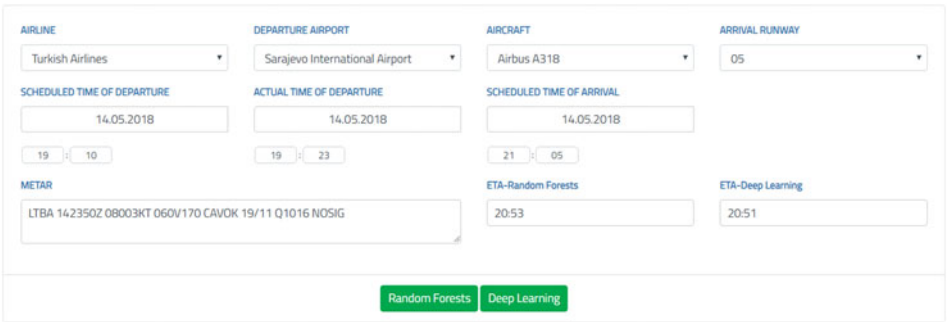


Figure 9. ETA prediction on the application.

of arrivals spread a very wide range around zero, where the distribution of errors is right-sided, which means that late arrivals are much greater in number than early arrivals.

Machine learning models were also used to predict the ETA after departure for another airport (Istanbul Sabiha Gokcen International Airport – ISG) with similar runway configuration. Table 7 shows the evaluation metrics for ISG. Both models succeeded to predict ETAs with a MAE less than 9min for an airport which was not included in the training process at all.

Machine learning models were obtained as 'h2o MOJO (Model ObJect, Optimized) models', which are intended to be easily embeddable in any Java environment. Then, the MOJOs were integrated into a Java web application developed for ETA prediction. Both models are able to predict the ETA in less than 5s. Figure 9 shows a screenshot of the ETA prediction for a sample flight where the actual time of arrival for is 20:55 UTC and scheduled time of arrival is 21:05 UTC. Thus, ETAs for any flights can be predicted dynamically by providing the required feature values.

5.0 CONCLUSIONS

In this study, machine learning models for ETA prediction were developed using RF and DNN. Both methods succeeded to predict the ETA with a MAE less than 6min immediately after departure and less than 3min immediately after TMA entrance. The prediction errors decreased 64% after departure and 84% after TMA entrance with respect to the scheduled time of arrival. Error histograms for ETA predictions show that most of the error is concentrated around 0, which indicates an exact prediction. The machine learning models were integrated into an application that predicts the ETA, given the feature values. The prediction takes less than 5s in application. The deep neural network model is 40% faster and 90% smaller in size compared with the random forest model.

https://doi.org/10.1017/aer.2021.13 Published online by Cambridge University Press

The machine learning models can be applied to airports with similar runway configurations without any further modifications because they use the flight distances rather than the arrival and departure airport pairs as categorical features. Flight distances can be calculated using a distance measure between departure and arrival airports. Therefore, having the features for a flight that landed with a runway configuration similar to the configuration of Istanbul Atatürk International Airport (such as ISG), ETA can be predicted without modifying the machine learning models.

The departure traffic, airport and airspace congestions will be integrated into the training process in a future work.

ACKNOWLEDGEMENTS

This research received no specific grant from any funding agency in public, commercial or non-profit sectors.

REFERENCES

1. EUROCONTROL. Challenges of Growth 2013 Task 4: European Air Traffic in 2035, 2013.
2. FEDERAL AVIATION ADMINISTRATION. FAA Aerospace Forecast: Fiscal Years 2018-2038, 2018.
3. AYHAN, S., COSTAS, P. and SAMET, H. Predicting estimated time of arrival for commercial flights, Proceedings of the 24th International Conference on Knowledge Discovery and Data Mining (KDD 2018), 2018.
4. KERN, C.S., MEDEIROS, I.P. and YONEYAMA, T. Data-driven aircraft estimated time of arrival prediction, Systems Conference (SysCon), 2015.
5. GLINA, Y., JORDAN, R. and ISHUTKINA, M. A tree-based ensemble method for prediction and uncertainty quantification of aircraft landing times, Proceedings 10th Conference on Artificial and Computational Intelligence and its Applications to Environmental Sciences, 2012.
6. SLATTERY, R. and ZHAO, Y. Trajectory synthesis for air traffic automation. *J Guidance Cont Dyn*, 1997, **20**, (2).
7. MONDOLONI, S. Aircraft Trajectory Prediction Errors: Including a Summary of Error Sources and Data, FAA/Eurocontrol Action Plan 16, 2006.
8. MURPHY, J.R., CLAYTON, J., REISMAN, R.J. and WRIGHT, R. Physics-based and parametric trajectory prediction performance comparison for traffic flow management, AIAA Guidance, Navigation, and Control Conference and Exhibition, 2003.
9. BREIMAN, L. Random forests. *Mach Learn*, 2001, **45**, (1), pp 5–32.
10. TAKACS, G. Predicting flight arrival times with a multistage model, IEEE International Conference on Big Data, 2014.
11. WANG, Z., LIANG, M. and DELAHAYE, D. A hybrid machine learning model for short-term estimated time of arrival prediction in terminal manoeuvring area. *Transport Res Part C Emerging Tech*, 2018, **95**, pp 280–294
12. BREIMAN, L., FRIEDMAN, J., STONE, C.J. and OLSHEN, R.A. Classification and Regression Trees, 1984.
13. DIETTERICH T. An experimental comparison of three methods for constructing ensembles of decision trees: bagging, boosting and randomization, *Mach Learn*, 2000, **40**, pp 139–157.
14. BREIMAN, L. Bagging predictors, *Mach Learn*, 1996, **26**, pp 123–140.
15. GHATAK, A. *Machine Learning with R*, Springer, Singapore, 2017.
16. ALPAYDIN, E. *Introduction to Machine Learning*, The MIT Press, Massachusetts, 2010.
17. DECHTER, R. Learning while searching in constraint-satisfaction-problems, Proceedings of AAAI-86, 1986, pp 178–183.
18. AIZENBERG, I.N., AIZENBERG, N.N. and VANDEWALLE, J. Multiple-Valued Threshold Logic and Multi-Valued Neurons, *Multi-Valued and Universal Binary Neurons.* Springer, 2000, pp 25–80.
19. LECUNN, Y., BENGIO, Y. and HINTON, G. Deep learning. *Nature*, 2015, **521**, pp 436–444.
20. H_2O.ai, https://www.h2o.ai/ (accessed 13 October 2020)

https://doi.org/10.1017/aer.2021.13 Published online by Cambridge University Press