

RESEARCH

Open Access



# Optimization of air traffic management efficiency based on deep learning enriched by the long short-term memory (LSTM) and extreme learning machine (ELM)

Mahdi Yousefzadeh Aghdam<sup>1</sup>, Seyed Reza Kamel Tabbakh<sup>2\*</sup> , Seyed Javad Mahdavi Chabok<sup>2</sup> and Maryam Kheyraadi<sup>1</sup>

\*Correspondence:

Drkamel@mshdiau.ac.ir

<sup>2</sup> Department of Computer

Engineering, Mashhad

Branch, Islamic Azad

University, Mashhad, Iran

Full list of author information

is available at the end of the article

## Abstract

Nowadays this concept has been widely assessed due to its complexity and sensitivity for the beneficiaries, including passengers, airlines, regulatory agencies, and other organizations. To date, various methods (e.g., statistical and fuzzy techniques) and data mining algorithms (e.g., neural network) have been used to solve the issues of air traffic management (ATM) and delay the minimization problems. However, each of these techniques has some disadvantages, such as overlooking the data, computational complexities, and uncertainty. In this paper, to increase the air traffic management accuracy and legitimacy we used the bidirectional long short-term memory (Bi-LSTMs) and extreme learning machines (ELM) to design the structure of a deep learning network method. The Kaggle data set and different performance parameters and statistical criteria have been used in MATLAB to validate the proposed method. Using the proposed method has improved the criteria factors of this study. The proposed method has had a % increase in air traffic management in comparison to other papers. Therefore, it can be said that the proposed method has a much higher air traffic management capacity in comparison to the previous methods.

**Keywords:** Air traffic management, LSTM, ELM, Deep learning

## Introduction

Air traffic management (ATM) refers to the required activities for the efficient and safe management of the national air system (NAS) in each country. Generally, ATM encompasses the two components of air traffic control (ATC) and air traffic flow management [1]. The ATC system mainly utilizes tactical decisions (e.g., real-time separation method) for collision detection. The NAS is divided into several sections to present ATC services and assist air traffic controller operators in the process of traffic control and flight separation by ATCs. The air traffic control methods for the prevention of flight delay and interference is a significant issue in the operating field of ATM [1]. Additionally, fleet fuel

and flight delays in airports and secondary costs impose significant multiple charges to the airlines.

The arrival schedule of flights is considered by airlines and aviation/airport companies [2], attracting attention to the airports with better ATM performance by airlines. Another challenge of flight control is several airports' coverage by a single ATM while each airport may have several pattern areas for its ATM [3]. Moreover, the traffic or pattern areas in nearby airports may be dependent or independent, and each airport may have several parallel or non-parallel runways. In this regard, the traffic of parallel runways may be dependent or independent, whereas crossover runways' traffic is dependent. Landing and takeoff runways might differ in every airport, or they might be jointly utilized [3, 4]. Furthermore, each runway may have several landing and takeoff procedures, which might have dependent and independent traffic. These issues demonstrate the high complexity of the problem modeling. Considering the huge scale of air traffic data (large data amount) in the classification learning process, the complexity level is higher with the increased number of categories of each class. Additionally, the selection of the significant features by traditional data mining approaches is almost impossible.

Various measures have been taken to solve the problems of ATM and ASP [5]. Many studies have aimed to solve these issues using mathematical models and methods, linear programming, mixed planning, and statistical models. However, one of the limitations of these studies is not considering the actual data and operating environment. Therefore, the proposed solutions lack the required accuracy and efficiency in the actual environments of airport aviation operations [6–17]. Some scholars have attempted to address the issue of traffic control and delay only by considering climatic and environmental conditions [10, 18–20]. Other studies have used the first come first serve (FCFS) technique, along with the queue model and other mixed methods, to solve the problem [21–25].

Another category of research and articles have applied data mining methods to investigate the influential factors in air traffic and flights [19, 26–30]. The machine learning algorithm is a conventional method which used to resolve the issues of air traffic, ASL, delay forecast, and minimization.

Machine learning is a branch of artificial intelligence or data analytics that deals with the development of algorithms that can be configured to learn from previous information. Machine learning is indeed a computational method for data mining [31, 32].

Overall, this technique has shown better problem-solving efficiency compared to other methods [33–37]. Some studies have also supported the uncertainty and fuzzy states, incorporating the latter into other techniques [38–40]. Such solutions have been proposed for ATC problems and are based on the Internet of Things (IoT) [16, 41, 42], optimization methods [6, 10, 12, 43], multi-objective optimization techniques [44–46], and intelligent agent-based methods [47].

An optimal approach to solving the mentioned issue involves using the structure of artificial neural networks [25, 43, 48]. Following the evolution of neural networks, deep learning networks have been considered to be one of the most recent and complete solutions in this regard. This novel technique could solve problems with a high accuracy owing to its ability to accept the large data of the problem and neural network integration, as well as learning techniques and structural dynamism, in the hidden layers formation. Aviation and ATM issues are no exception, and most of the recent studies

regarding flight delay forecast and flight control traffic have benefited from this technique [1, 47, 49–51].

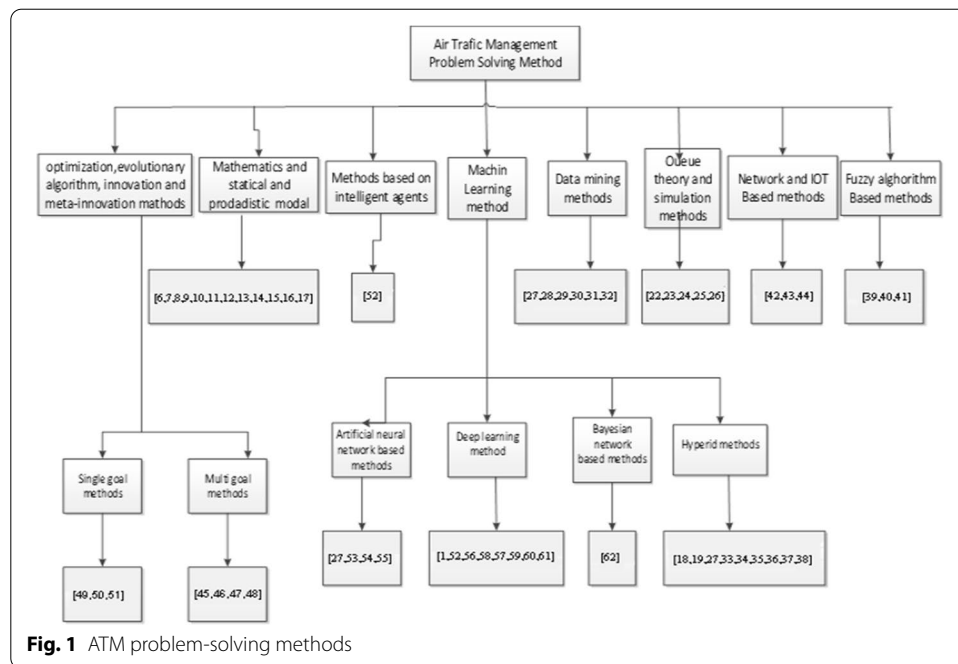
Multilayer neural networks or deep neural networks are included in machine learning subject and a set of algorithms that attempt to model high-level abstract concepts based on learning at various levels and layers, thereby enabling deep learning to process large volumes of data in complicated categories [1, 49, 50, 52–54]. In the current research, it was attempted to propose an accurate and proper method to solve the problem within the operational domain of the terminal management area (TMA) using a combination of a deep neural network and other methods. Furthermore, the present study is aimed to propose a deep learning model using a long short-term memory (LSTM)-based deep learning model and recurrent neural network (RNN) in order to increase the predictive accuracy of short and long-term annual windows by enhancing deep learning (two-dimensional). In the third phase, the output of the deep model was transferred to the extreme learning machine (ELM) and fast learning deep neural machine in order to calculate the estimated time of arrival (ETA) and estimated time of departure (ETD) of each flight based on other similar input data, including the NAS data, bureau of transportation statistics (BTS) system, and automatic dependent surveillance-broadcast (ADS-B) system. Ultimately, an flights control was developed within the airport TMA range with a 15-min time window for flight arrival using evolutionary and meta-heuristic algorithms by conforming the flight rules to the learning outcomes and increasing the accuracy.

The following sections of the article have been structured, with section two reviews the previous methods in the ATM field, section three describes the proposed model, and section four, evaluates and compares the results with other techniques; in addition, section five has been dedicated to a conclusion.

### **A review of previous methods**

Numerous efforts have been made to solve the ATM problem and minimize the rates of ETA and ETD delay in various dimensions. Most of the studies in this regard have evaluated inbound and outbound flights separately, attempting to propose solutions using methods such as data mining and mathematical techniques (Fig. 1).

In [1], researchers used deep learning architectures such as stacked auto encoders, convolutional neural networks, and recursive neural networks as the architecture for forecasting daily delay status. The aim of the present study was to estimate the daily delay at each airport and calculate the delay for a specific flight based on the obtained results. In order to forecast the daily delay, we initially calculated the mean delay of the inbound and outbound flights and added the estimated value to a recursive neural network, along with the weather data as a sequence, which was added to the output after determining the weight and bias of the separate data. Weighting and biasing are repetitive procedures, and each replication stage determines the value of cost function using the stacked memory cell structure (LSTM) and Sigmoid and Tanh functions were replaced by the structure of the RNN. As a result, the information of each hidden layer was stored, thereby increasing the model efficiency. However, the proposed method limitation was the elimination of the details from the rounds related to the management of



the ground delays for the flight preparation of the aircrafts. Another issue was the lack of a deeper LSTM structure in a time-based forecast structure, which increased accuracy.

In [26], the main objective was to minimize the estimated delay along with the sequence of the flights in resolving the ATM problem; a combination of clustering and neural network was used to solve the problem. The integrated technique had two steps of clustering-based forecast and multi-cell neural network (MCNN)-based forecast. In the first step, principal component analysis (PCA) was used to reduce the variable dimensions of the path vector after the clearing, filtering, and re-analyzing of data. Afterwards, the paths were clustered into several patterns with a clustering algorithm. In the forecast phase, the MCNN model was applied to predict the four-dimensional (4D) aspect of the density path. In addition, there was a predictor for each path partition, which encompassed an NN-based learning cell. Each exclusive learning cell was trained by a set of related paths, and each paths set included the related prediction model. According to the obtained results, the proposed model in present study was stronger, more accurate, and more efficient for short-term predictions. However, some of its limitations were the low-data scale and lack of highly accurate learning methods (e.g., deep machine learning). These challenges were eliminated by using a large data volume and the deep learning technique.

Researchers have used a combination of deep belief networks (DBNs) and PCA to evaluate the aviation safety of the country. In general, DBN is useful for safety prediction since each layer acquires more complicated features from the previous layers. In fact, DBN predicts severe flight incident rates based on PCA results [49]. In this case, we assessed seven main factors for unsafe events systematically and in detail, including aircraft, landing and takeoff, aircraft operation, airport and aircraft, ground transportation, and weather. According to the obtained results, the predicted PCA-DBN data were

compatible with the actual data on flight incidents. In this regard, the proposed model was considered superior compared to the gray neural network, support vector machine, and DBN. In [41], the researchers proposed a deep learning-based model to predict the hard landings of aircrafts based on the quick access record (QAR).

In IoT environment, devices collect data, which are sent to the IoT open cloud platform to be processed and analyzed. The prediction of a hard landing is a common application of IoT in the aviation field. Initially, 15 aircraft landing sensor data were selected from 260 parameters based on meteorology. Afterwards, an LSTM-based deep prediction model was developed to predict hard landing incidents using the selected sensor data. The empirical results showed high performance owing to the increased accuracy of the QAR data. The proposed model in this research was accurate and efficient in the prediction of hard landings, which guarantees the passengers safety and the decreased rate of flight incidents.

In [39], the main goal was to detect traffic accidents in order to increase road safety using the deep learning algorithm and stacked auto encoder model. In addition, the back propagation algorithm was used for the accurate adjustment of the parameters in the deep network. Ultimately, a fuzzy controller was exploited to increase the output accuracy of the deep network and adjust the neural network learning parameters based on the mean squared error (MSE). According to the findings, the fuzzy logic systems could be suitable for uncertain or approximate reasoning and allowed decision-making with the estimated values in incomplete or uncertain information.

In [28], the researchers presented various methods of data mining in the air transport lounge and assessed their efficacy. The proposed methods were assessed in three types of air transport data, and the flight recording information was provided by a flight recorder for the first time. The unofficial flight information recorder is known as the black box. In an aircraft equipped with a flight recorder, usually up to 500 variables of information are recorded per second for the flight duration, such as time, altitude, vertical acceleration, and vertex. While some of these variables are distinct, the others may be continuous. The artificial data were the second type of the aviation information. This information were focused on flight anomalies. This concept intentionally embedded in the data to examine the ability of the algorithm to detect the anomalies.

These anomalies which might be an unusual sequence of events or an unusual period between events. The second type of data is aviation crash reports, which have no strict rules, and the pilot needs no specific conditions as the reports include narrations. However, a method should be designed to determine the significant data due to their lack of unity.

As for labels and labeled data in aviation data mining, a label is a descriptive word allocated to data based on specific features. In the present study, labels were considered a factor for the formation of a flight incident.

Some of the factors cause to flight incidents included diseases, hazardous environments and autopilot. To improve the accuracy of flight characteristics, the researchers used the time warp edit distance (TWED) and k-means algorithms [29]. In first, the researchers assessed a dataset of flights with the desired time in the case of flight routing with the same origin and flight destination to eliminate the effect of the exit point. Then, the adapted k-means algorithm was proposed, in which the distance

between various paths was estimated by the TWED algorithm rather than the conventional elastic similarity measurement in the k-means algorithm. In this research, one of the benefits of the proposed method was the increased accuracy of the algorithm and higher efficacy of using the controlled airspace in air traffic management. On the other hand, one of the key limitations of the method was not considering the large scale of the data and use of large data, which led to the higher accuracy of the algorithm.

In [55], the researchers proposed a hybrid method of Bayes method and Gaussian mixture model–expectation maximization algorithm (GMM)–EM algorithm to predict and analyze the influential factors in the delay of the flights in Brazil aviation routes. Initially, the degree of the impact of each factor was calculated using outdated data. Then Bayes rules at specific points of the flight route followed by determining whether the delays occurred in larger domains. The next stage involved the estimation of the probability of the delays using the GMM–EM and EM algorithms, which are based on similarity in data. According to the obtained results, the probability of the delays at high levels could be predicted by determining the factors at low levels. Moreover, the GMM–EM algorithm could find more values for the similarity function compared to the EM algorithm, thereby reaching convergence sooner. Moreover, the accuracy of the model was observed to increase, which in turn improved the reliability of the prediction results.

In [8], the researchers focused on real-time aircraft routing and planning. In a crowded traffic control area (TCA), problems occur in case of traffic, which is specifically challenging for TCA operation management due to the growing demand for traffic, and the TCAs turn into the bottleneck of the entire ATC system. In this research, the method of linear programming formulations along with flight safety rules has been used to solve the problem, in order to minimize the maximum delay in the entire travel time due to the potential of aircraft congestion. Computational tests have been performed on real data from Rome Airport, the largest airport in Italy in terms of passenger demand.

The solution provides the optimal compromise among various objectives. In [9], the researchers proposed a new, efficient computational algorithm to resolve the uncertainty of the air traffic follow management using a limited probability optimization method. They initially developed a chance-constrained model based on the previous integral planning optimization model of the ATFM for the limitation of the possible capacities of the section. Afterwards, a polynomial approximation-based approach was applied to manage the chance optimization problems at large scales. One of the benefits of the proposed method was considering the uncertainty states in ATM, while the main limitation of the technique was the lack of using deep learning methods for large data in order to obtain a more accurate model.

In [23], the researchers estimated the input delay time and number of the aircrafts entering a controller space at a single time using a queue model and regression function, while also considering climatic conditions. In addition, the delay was forecasted before reaching the destination by considering variables such as the type of the aircraft, time of arrival, and times of entering and exiting the control space. The overall results of the optimization and artificial intelligence-based operation methods demonstrated that the artificial intelligence methods could overlook some of the errors, which rendered them

extremely more accurate compared to the queue models. Meanwhile, the queue model and recursive neural network were observed to have higher learning levels.

In another study, a 3-D convolutional neural network (R-3D CNN) was applied to increase the accuracy of air traffic predictive accuracy [56]. Changes in spatial-temporal air movements could be comprehensively considered by using this algorithm. In the mentioned study, the traffic situation graphics (TSG) sequence was applied to extract the prominent features. The proposed TSG enabled the consideration of some real-time factors to enrich the input information. As such, the model input was determined by combining the traffic situations at various light levels with the areas that were specified by other real-time factors, such as important tasks and public air traffic lights. The length of the input sequence was set to 30, 60, and 90 min before the prediction moment in order to determine the effect of the temporal dependencies, so that the optimal architecture proposed could be selected. Furthermore, the evaluation of the prediction results along with the three statistical factors confirmed the ability of the proposed model to yield accurate and sustainable predictive results for the air traffic system by distribution at various optical levels.

In [57], the main objective was to predict flight routes using a deep neural network in a capacity management and air traffic operational system. A deep neural network was trained in the historical routes and a set of predictors, and the neural network predicted the most likely route through the airspace. In addition, the network was able to generalize the results to the flights and conditions that had not been detected before. The neural network could also prevent changes by repetitive educations on the newly recorded data. In the mentioned study, an integrated solution was used in the air traffic platforms with the capacity for 10% of the total traffic, and the results of the solution showed the level of the apparent progress.

The promotion of user confidence increases the domain of all traffic.

Large European airports consider strategic flight plans to reduce the air traffic capacity that demand imbalance. In these airports, flights are assigned an entry or departure slot a few months before takeoff. In this regard, the researchers in [58] evaluated such strategic plans using the predictions of the flight delays arriving, departing or canceling. The proposed approach was used in London Heathrow Airport during 2013–2018, and the resulting flight plan was assessed in terms of the predicted flight cancelation and delay using a machine learning approach. According to the findings, the proposed method was able to provide the airport coordinators of the possible delays and cancelations related to the strategic plans. In [59], an end-to-end deep learning-based approach was also presented to increase accuracy in the air traffic flow using the CNN and RNN algorithms, as well as a convolutional LSTM module, which was proposed to construct a trainable model to predict the air traffic flow. The experimental results of the actual data were indicative of the superior performance compared to the current approaches used for predictive accuracy and stability.

Moreover, the proposed model could predict the flow distribution at various flight levels in the flights controlled space, which in turn improved the ATM level. The analysis of the distribution of the prediction errors on various spaces cells, flight levels, and prediction of the samples indicated that the spatial and temporal transmission patterns of the flight flow in the ATM system could be thoroughly learned by the proposed model. On

the other hand, the proposed model could predict that the optimal air traffic management measures were taken to improve performance efficacy system.

In [17], the researchers used bidirectional long short-term memory (BLSTM) for the performance data of air transportation management in order to identify the system. In the system, BLSTM was able to reconstruct the nonlinear temporal series and make valid predictions. According to the other findings of the mentioned study, neural networks in deep learning methods could manage complicated nonlinear temporal series and learn to reconstruct these series based on multidimensional inputs, while also storing their knowledge regarding the behavior of the observation datasets.

In [60], a multi-step deep sequence learning model (Bi-LSTM + Seq2Seq) was proposed to predict airport delay based on the spatial and temporal relations of the other airports within the network. In the first step, the dataset was processed for the analysis the correlation between the temporal delays of the airports based on the complex network theory. Afterwards, the PageRank and K-means algorithms were applied to cluster the behavior of the networks and identify their overall status. At the next stage, the Bi-LSTM + Seq2Seq model was proposed and trained based on the time-series data on the current status of the network and delay in the interactions between the airports. The experimental results indicated that the suggested model had higher accuracy and sustainability compared to other prediction algorithms.

In [31], the main objective was proposing a deep learning-based method to evaluate the delays in inbound flights. Initially, the important features were extracted, followed by model training by artificial neural networks and DBN using random samples. In the mentioned study, researchers applied the momentum learning rate and resilient back propagation, which acted extremely quicker than back propagation, thereby increasing the training pace and model convergence. Notably, the DBNs were based on a Boltzmann machine, where each layer received communications from the previous layer, and a Boltzmann machine was added to the network at each stage. During the training, the inaccurate classification error rate decreased by the fine-tuning of the parameters and momentum learning rate. Since the output of each layer was divergent, the training pace decreased, and the gradient tended to zero.

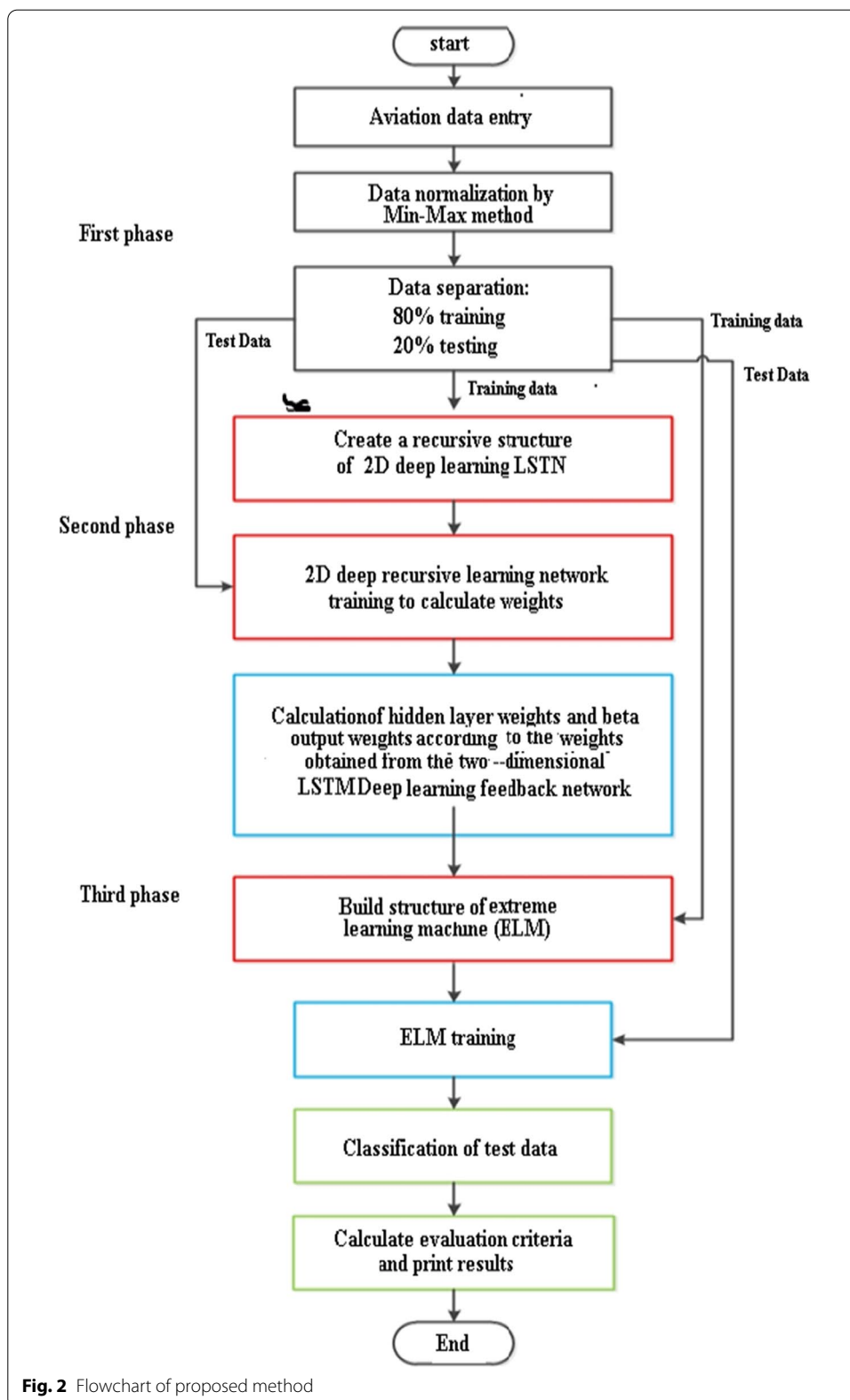
### **Proposed method**

The proposed model in the present study was based on the LSTM and ELM algorithms. Figure 2 depicts the flowchart of the proposed method. As is observed in Fig. 2, the suggested method had three phases of the loading, normalization, and separation of the data, creating a two-dimensional LSTM back learning structure using a Bi-LSTM neural network, while also estimating the beta weights, training the ELM, and calculating the assessed criteria. The proposed steps have been further explained.

#### **Uploading, normalization, and separation of the data**

At this phase, the dataset obtained from Kaggle.com [53], which contained 100,000 records and five features, was uploaded, and the Min–Max normalization approach, as is shown in Eq. 1, in order to facilitate the comparison of the results.





**Fig. 2** Flowchart of proposed method

$$x_{norm} = \frac{x - x_{min}}{x_{max} - x_{min}} \quad (1)$$

In the equation above,  $X_{min}$  and  $X_{max}$  are the minimum and maximum of the main feature, respectively,  $X$  represents the value of the main feature, and  $X_{norm}$  is the normalized feature value.

### Creating a two-dimensional LSTM back learning structure and using a Bi-LSTM neural network

At this stage, the initial net weights of the ELM neural network were created using the neural network structure with Bi-LSTM. GRU and LSTM have the same function, which is to find out whether there is a long-term dependency and to overcome the problem of vanishing and exploding gradient. LSTM does it through three gates, namely a forget gate that controls how much information needs to be removed, an input gate that controls how many cell states need to be stored, and an output gate that controls how many cell states are sent to the next cell have to [61, 62].

The LSTM network architecture has been initially developed by Hochreiter and Schmidhuber [31, 60]. In this structure, the input sequence vector of  $x = (x_1, x_2, \dots, x_n)$  was provided, where  $n$  represented the sentence's length. The primary structure of the LSTM was based on the use of three control gates to control a memory cell activation vector. The first forget gate determined that the value of the  $c_{t-1}$  cell at the previous time was maintained until the time of the current cell status of  $C_t$ . The second input port determined the amount of the  $x_t$  storage of the network input to the  $C_t$  of the current state of the cell, and the third output gate determined to what extent the  $C_t$  was transferred to the current output value of the LSTM networks. The three gates were an entirely connected layer, the layer's input was a vector, and the output was an actual number. In Fig. 3, the initial structure of the LSTM cell has been demonstrated, which is interpreted as follows:

In the equation above,  $\sigma$  is the sigmoid logistic function,  $x_t$  shows the  $t$ -th word vector of the sentence, and  $h_t$  is the latent state. In addition,  $W$  and  $b$  demonstrate the

$$\begin{aligned} \text{Input gates : } i_t &= \sigma(W_{ix}x_t + W_{ih}h_{t-1} + b_i) \\ \text{Forget gates : } f_t &= \sigma(W_{fx}x_t + W_{fh}h_{t-1} + b_f) \\ \text{Output gates : } o_t &= \sigma(W_{ox}x_t + W_{oh}h_{t-1} + b_o) \\ \text{Cell states : } c_t &= f_t * c_{t-1} + i_t * \tanh \cdot (W_{cx}x_t + W_{ch}h_{t-1} + b_c) \\ \text{Cell outputs : } h_t &= o_t * \tanh(c_t) \end{aligned} \quad (2)$$

weight matrices (e.g.,  $W_{xt}$  is the weight matrix of the forget gate) and bias vectors (e.g., the input gate bias vector), respectively for the three input gates. In order to overcome the shortage of a single LSTM cell, which could only record the previous fields but does not use the future field, two hidden LSTM layers were combined with the same output separately from different directions in the BRNN neural networks. By this structure, the output layers were able to apply the related information from both the previous and future cases.

Moreover, BiLSTM calculated the input sequence of  $x = (x_1, x_2, \dots, x_n)$  from the opposite direction to the hidden sequence forward of  $h_t^{\rightarrow} = (h_1^{\rightarrow}, h_2^{\rightarrow}, \dots, h_n^{\rightarrow})$  and hidden

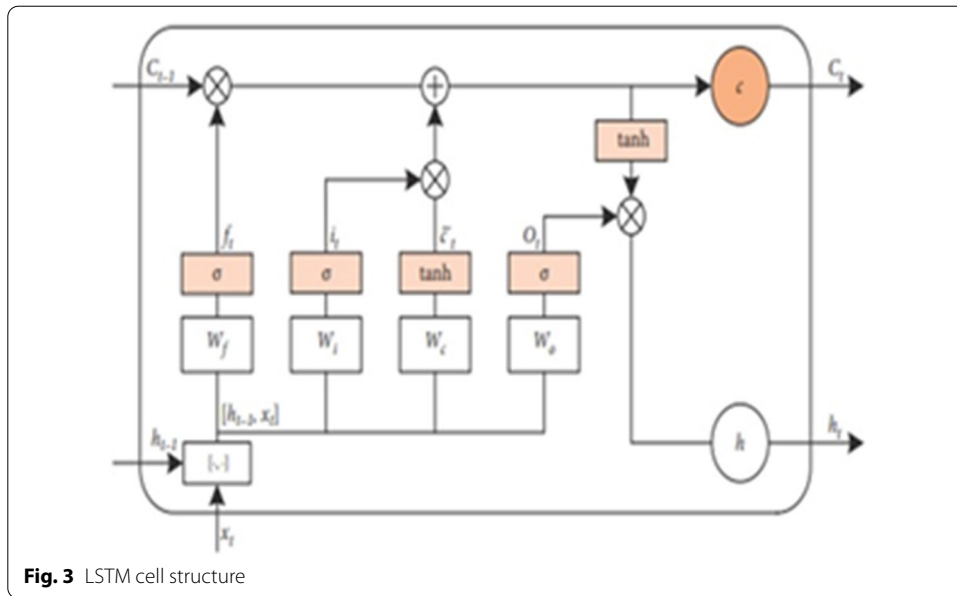


Fig. 3 LSTM cell structure

sequence backward of  $(h_t^{\leftarrow} = (h_{t-1}^{\leftarrow}, h_{t-2}^{\leftarrow}, \dots, h_1^{\leftarrow}))$ . The encoded vector of  $y_t$  also encompassed an aggregation of the final forward and outward outputs.

$$\begin{aligned}
 y_t &= [h_t^{\rightarrow}, h_t^{\leftarrow}] \\
 h_t^{\rightarrow} &= \sigma(W_{h \rightarrow x} x_t + W_{h \rightarrow h} h_{t-1}^{\rightarrow} + b_{h \rightarrow}), \\
 h_t^{\leftarrow} &= \sigma(W_{h \leftarrow x} x_t + W_{h \leftarrow h} h_{t-1}^{\leftarrow} + b_{h \leftarrow}), \\
 y_t &= W_{y h \rightarrow} h_t^{\rightarrow} + W_{y h \leftarrow} h_t^{\leftarrow} + b_y,
 \end{aligned}
 \tag{3}$$

In the equations above,  $y = (y_1, y_2, \dots, y_t, \dots, y_n)$  is the output sequence of the first hidden layer. Some of the findings in this regard have suggested that classification or regression performance could be further improved by accumulating multiple

$$L = H_A^T H_Q \in R^{m \times n}
 \tag{4}$$

BiLSTMs in neural networks [60]. In addition, theoretical evidence suggests that a deep hierarchical model is more efficient in delivering some functions than the shallow type. In the present study, an accumulated BiLSTM network was defined, where the output  $y_t$  from the lower layer was converted into the input of the upper layer. The accumulated BiLSTM structure is shown in the following (Fig. 4):

$$h_t = W_{hh} h_t^{\rightarrow} + W_{hh} h_t^{\leftarrow} + b_h,
 \tag{5}$$

The definition of  $A = (a_1, a_2, \dots, a_m)$  and  $Q = (q_1, q_2, \dots, q_m)$  show the sequence of the problem and sequence of the responses, respectively, where  $n$  and  $m$  demonstrate the length of the problem and responses,  $a_t$  and  $q_t$  are the  $t$ -th words of the problem and responses. In this section, the stacked BiLSTM was implemented on the problem and Fig. 4. Structure of stacked BiLSTM networks responses, and the hidden-mode  $H_Q$  and  $H_A$  matrices were obtained.

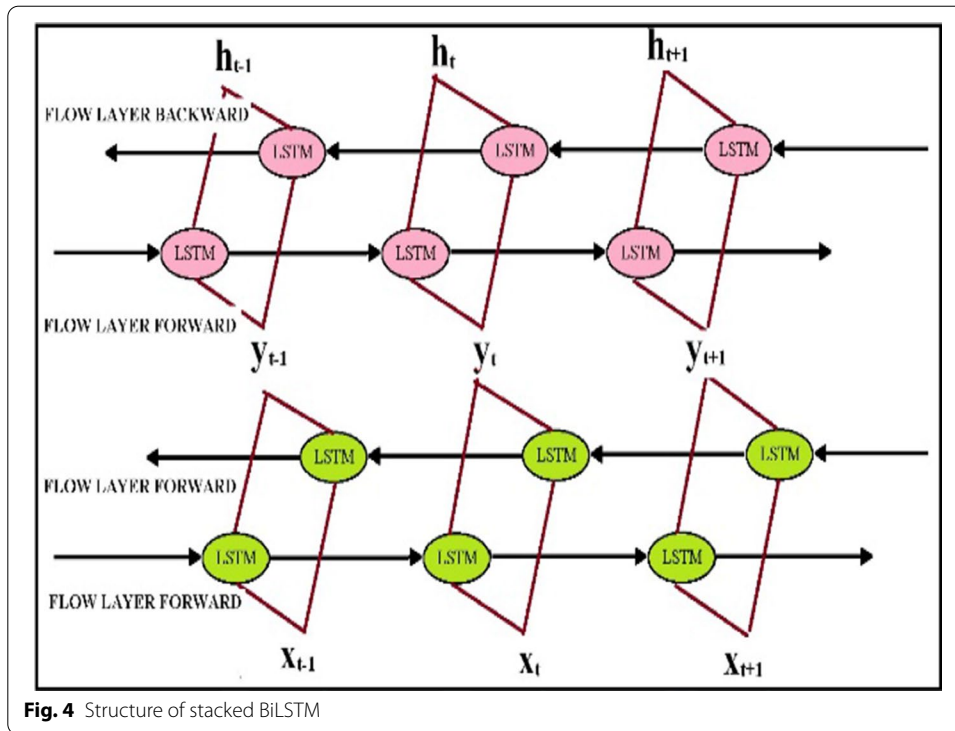


Fig. 4 Structure of stacked BiLSTM

$$\begin{aligned}
 h_t^q &= \text{sBiLSTM}(h_{t-1}^q, h_{t+1}^q, q_t), h_0^q = 0, \\
 h_t^a &= \text{sBiLSTM}(h_{t-1}^a, h_{t+1}^a, a_t), h_0^a = h_n^q,
 \end{aligned}
 \tag{6}$$

The calculations are as follows:

$$\begin{aligned}
 H_Q &= [h_1^q, h_2^q, \dots, h_n^q] \in \mathbb{R}^{d \times n}, \\
 H_a &= [h_1^a, h_2^a, \dots, h_m^a] \in \mathbb{R}^{d \times m},
 \end{aligned}
 \tag{7}$$

where  $d$  represents the dimensions of the hidden mode (Fig. 5).

**Coherence mechanism for problem presentation**

In this section, a coherence mechanism was implemented to encode the problem in accordance with the response sequence (Fig. 6). We attempted to interact more closely with the functions and summaries in the coherence mechanism by designing the matrices' multiplication to address more questions. Initially, the matrix multiplication was carried out to estimate the  $L$  matrix, which included the propensity scores related to all pairs of the problem and response.

The Soft max function was used to standardize the vector elements and was likely to be effective against multiple classifications and distribution problems. Therefore, column-row-based Soft max functions were used to generate accuracy weights for the hidden modes of the problem and response separately by the following equation:

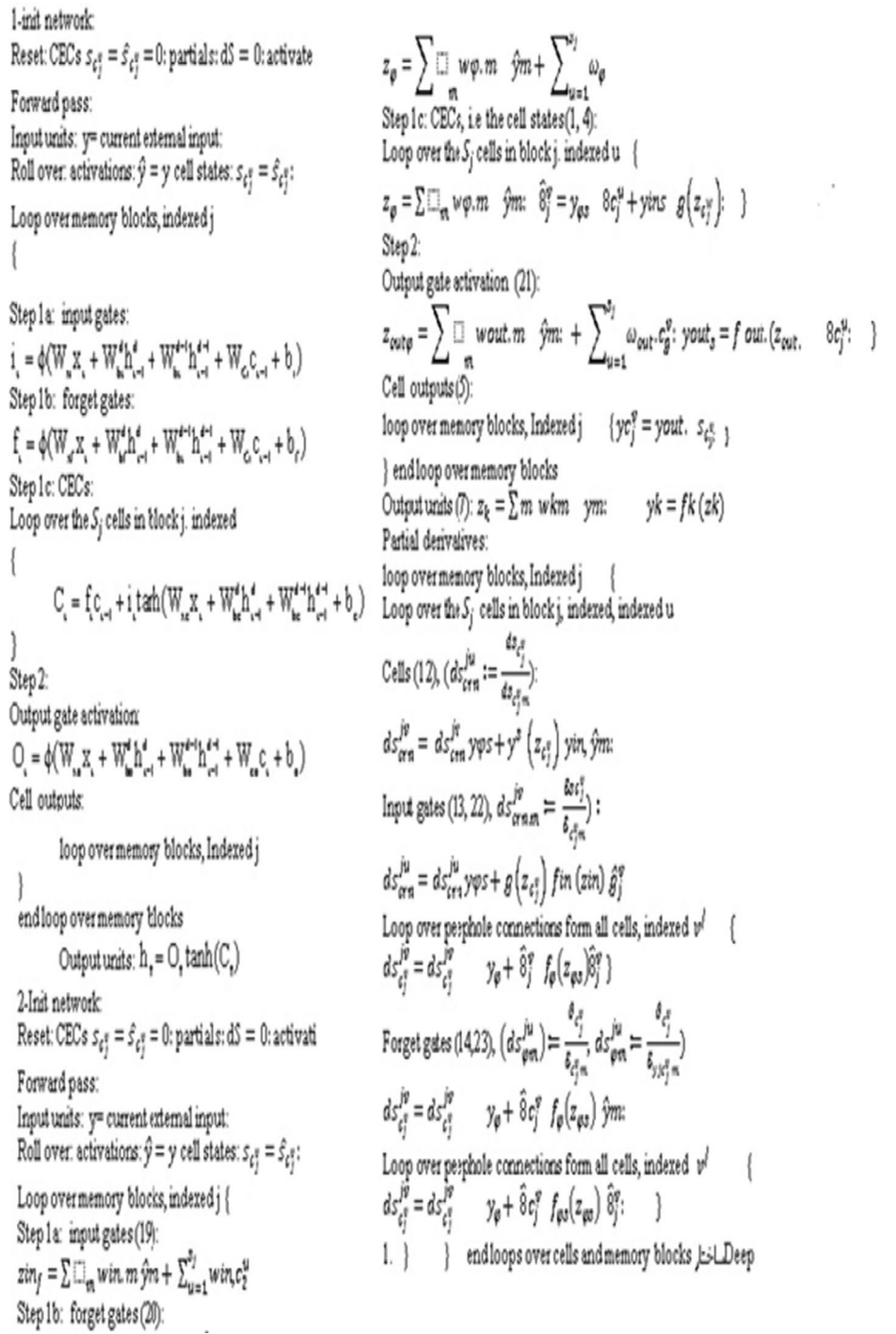
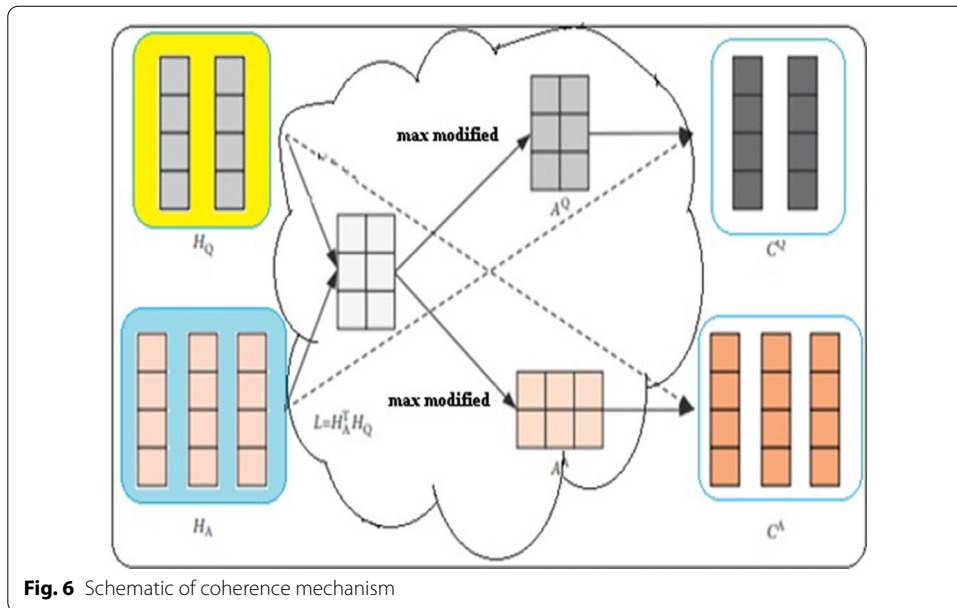


Fig. 5 Pseudocode of proposed blstm



**Fig. 6** Schematic of coherence mechanism

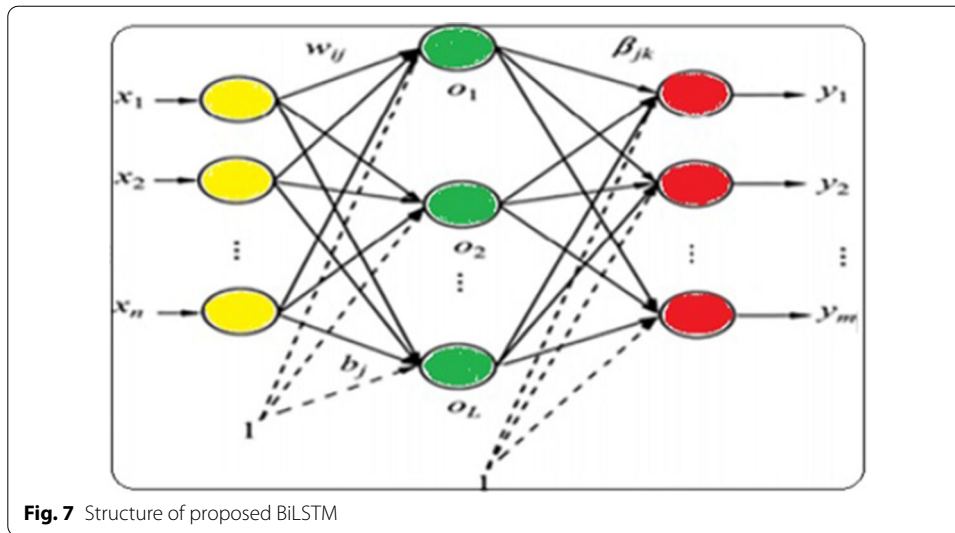
$$\begin{aligned}
 A^Q &= \text{softmax}(L) \in R^{m \times n}, \\
 A^T &= \text{softmax}(L^T) \in R^{m \times n},
 \end{aligned}
 \tag{8}$$

In order to obtain the accuracy vector of the question with respect to each word of the response, we combined the explanatory weights and approximation matrix to calculate the new  $C^Q$  and  $C^A$  field vectors. In this section,  $C^Q$  and  $C^A$  were the results of the interaction between the problem and vector response, as follows:

$$\begin{aligned}
 C^Q &= H_A A^Q \in R^{d \times n} \\
 C^A &= H_Q A^A \in R^{d \times m}
 \end{aligned}
 \tag{9}$$

**Attention mechanism (accuracy) to display the problem**

A soft accuracy layer could be used for the integration of information from the words of the problem and response in order to reduce the information loss of the stacked BiLSTM [60, 63]. In the proposed model, the attention mechanism was applied for the cohesion output. In the current research,  $C^Q$  was assumed to show the  $t$ -th attention field vector of this problem, and the maximum aggregation occurred to convert the input into a vector with  $O_q$  fixed length. In addition, the software weight of all the text vectors ( $CA, CA_2, \dots, CA_m$ ) could be learned independently based on  $O_q$  through the attention mechanism, and the  $O_a$  weight field vector used the response as the final representation.



$$\begin{aligned}
 O_q &= \max_{0 < t < n} C_t^Q, \\
 M_{aq}(t) &= \tanh(W_{am}C_t^A + W_{qm}O_q), \\
 S_{aq}(t) &\propto \exp(W_{ms}^T M_{aq}(t)), \\
 O_a &= \sum_{t=1}^m CA_t S_{aq}(t)
 \end{aligned} \tag{10}$$

In the equations above,  $W_{am}$  and  $W_{qm}$  show the attention matrices of  $C_t^A$  and  $O_q$ , respectively, and  $W_{ms}$  is the attention weight vector. The official presentation of the  $Q_a$  response was determined based on the attention (accuracy) weight of  $S_{aq}(t)$  for the  $t$ -th word response text vector. In addition, normalization occurred by the performance of the Soft max function, which was proportional to  $C_t^A$ . The higher values of  $S_{aq}(t)$  demonstrated a more significant correlation between  $C_t^A$  and the problem, while the problem vector drew more attention (Fig. 7).

**Calculation of beta weights and ELM training**

Compared to the BP networks, the ELM network lacks the output layer bias. While the input weight and bias of the hidden layer of the ELM network are generated randomly, the weights obtained from the neural network encompassing BiLSTM was applied at this stage of the present study, and only the output weight had to be determined, which could limit the manual adjustment of the parameters of each layer in the BP neural network and improve the predictive accuracy. Figure 8 depicts the structure of the ELM.

As can be seen,  $x_1, x_2, \dots, x_n$  were the input of the educational data, and  $w_{ij}$  and  $\beta_{jk}$  were the input weight in the neural network and indicative of the output weight vector between the hidden layer and output node, respectively. As a result, the output of the hidden layer corresponded to the  $x$  input. In this regard,  $OL$  was the node of the hidden layer, and  $b_j$  was the neuron threshold in the hidden layer. In addition,

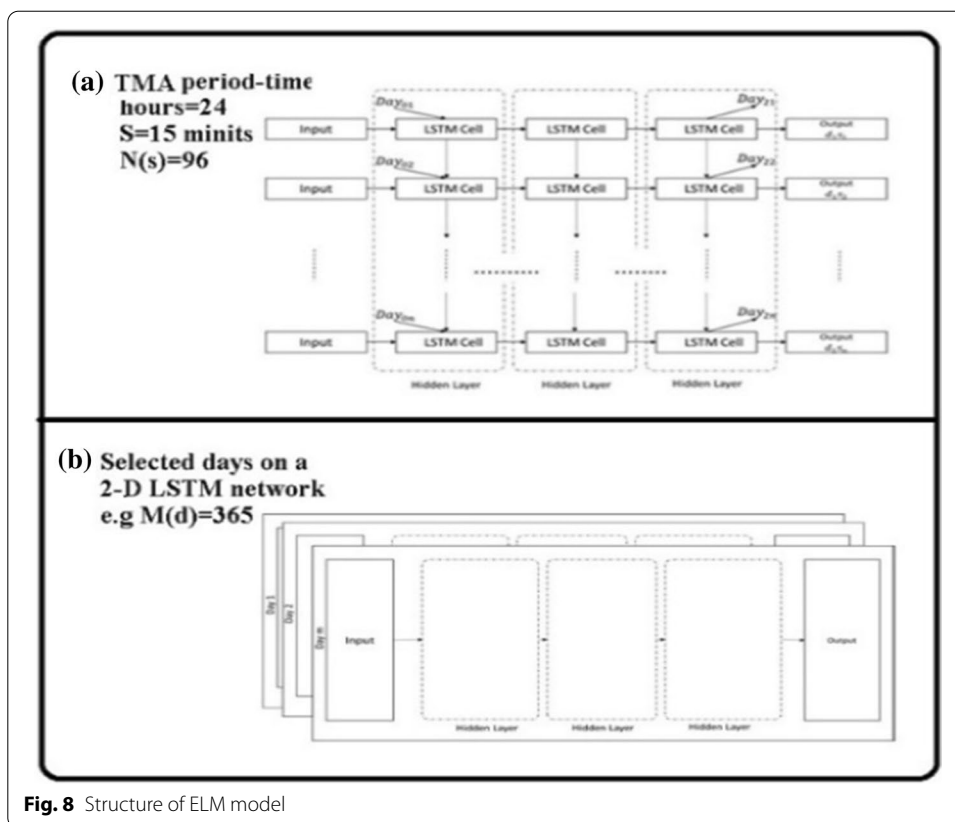


Fig. 8 Structure of ELM model

the education sample set was  $\{(x_i, y_i) | x_i \in R^n, y_i \in R^m, i = 1, 2, \dots, N\}$ , and the  $L$  hidden layer was the number of the neural cells. The excitation function was shown by  $g(x)$  in the ELM. In the current research, sigmoid was selected as the  $g(x)$  function, and the ELM model could be defined, as follows:

$$\sum_{i=1}^{\tilde{N}} \beta_i g_i(x_j) = \sum_{i=1}^{\tilde{N}} \beta_i g_i(w_i x_j + b_i) = o_j, j \in [1, N] \tag{11}$$

The matrix was equal to:

$$H\beta = Y \tag{12}$$

In the equation above

$$\beta = [\beta_1, \beta_2, \dots, \beta_L]^T_{L \times m} Y = [y_1, y_2, \dots, y_L]^T_{N \times m} = \begin{bmatrix} g(w_1 x_1 + b_1) & \dots & g(w_1 x_1 + b_L) \\ \vdots & \ddots & \vdots \\ g(w_1 x_N + b_1) & \dots & g(w_1 x_N + b_L) \end{bmatrix}_{N \times L} \tag{13}$$

Equation 14 was equivalent to minimizing squares, as follows:



**Table 1** Applied dataset

ROW	Feature	Description
1	ORIGIN_AIRPORT	The airport from which the flight takes off
2	DESTINATION_AIRPORT	The airport where the flight lands
3	AIRLINE	Flight Operations Airlines
4	MONTH	Month Flight date
5	DAY	flight date
6	DAY_OF_WEEK	Flight day (number per week)
7	FLIGHT_NUMBER	Flight number registered in the flight plan
8	DEPARTURE_TIME	Time of departure of the aircraft from the runway
9	SCHEDULED_TIME	Scheduled time for flight departure in flight schedule
10	ELAPSED_TIME	Predict the time for the plane to leave or take off
11	AIR_TIME	Flight time in the flight cruise section
12	DISTANCE	Flight distance between the airport of origin and the airport of destination
13	ARRIVAL_TIME	Flight landing time at the destination airport
14	ARRIVAL_DELAY	Flight delay rate in flight landing
15	DEPARTURE_DELAY	Flight delay in departure

$$\hat{\beta} = \operatorname{argmin}_{\beta} \|H\beta - |Y| \|_F \tag{14}$$

Equation 15 was solved as:

$$\beta = H^+T \tag{15}$$

The inverse  $H^+$  was generalized from the hidden output layer matrix. In the final step, we examined the assessable criteria.

**Analysis and evaluation**

At this stage of the research, we are analyzed and evaluated the applied data and assessed the results and criteria.

**Dataset**

The dataset obtained from [1, 54] included 1,00,000 records and 15 features according to Table 1.

The U.S. Department of Transportation’s (DOT) Bureau of Transportation Statistics tracks the on-time performance of domestic flights operated by large air carriers. Summary information on the number of on-time, delayed, canceled, and diverted flights is published in DOT’s monthly Air Travel Consumer Report and in these datasets of (2015 or 2005 till 2015 or 2010 till 2015 or 2005 till 2008) flight delays and cancellations. The flight delay and cancellation data was collected and published by the DOT’s Bureau of Transportation Statistics.

Each entry of the flights.csv file corresponds to a flight and we see multi version of this dataset in variable of times e.g. the dataset that recorded in 2015 have more than 5,800,000 flights. These flights are described according to 31 variables. A description of these variables as follow:

**Table 2** Confusion matrix

Real values	Predicted values	
	With delay	Without delay
With delay	TP (true positive)	FN (false negative)
Without delay	FP (false positive)	TN (true negative)

YEAR, MONTH, DAY, DAY\_OF\_WEEK: dates of the flight.

AIRLINE: An identification number assigned by US DOT to identify a unique airline.

ORIGIN\_AIRPORT and DESTINATION\_AIRPORT: code attributed by IATA to identify the airports.

SCHEDULED\_DEPARTURE and SCHEDULED\_ARRIVAL: scheduled times of take-off and landing.

DEPARTURE\_TIME and ARRIVAL\_TIME: real times at which take-off and landing took place.

DEPARTURE\_DELAY and ARRIVAL\_DELAY: difference (in minutes) between planned and real times.

DISTANCE: distance (in miles) An additional file of this dataset, the airports.csv file, gives a more exhaustive description of the airports.

**Table 3** Confusion matrix description

TP	The number of the behaviors that represented the existence of a delay and were correctly predicted by the model
FP	The number of the behaviors that represented the presence of a delay, and the model incorrectly predicted the absence of delay
FN	The number of the behaviors that indicated the absence of delay, and the model incorrectly predicted the existence of delay
TN	The number of the behaviors that showed the lack of delay, and the model correctly predicted them

**Table 4** Formulations description

$Accuracy = \frac{TP+TN}{TP+TN+FP+FN}$	This was the most important criterion for determining the performance of a classification algorithm, which showed the percentage of the proper classification of the total set of the experimental record
$Rcall = \frac{TP}{TP+FN}$	It showed the ability of the algorithm to accurately detect delay
$Specificity = \frac{TN}{FP+TN}$	It demonstrated the efficiency of the classifier in the accurate prediction of the lack of delay
$Precision = \frac{TP}{TP+FP}$	It demonstrated the ability of the algorithm to detect the positive categories (i.e., delay)
$F - measure = \frac{2*Recall*Precision}{Precision+Recall}$	It showed the harmonic mean between accuracy and recall
$RMSE = \sqrt{\sum_{t=1}^n (y - \hat{y})^2 / n}$	Measuring the accuracy of the predicted rates compared to the correct rates
$MSE = \sum_{t=1}^n (y - \hat{y})^2 / n$	It was a statistical tool to determine the predictive accuracy in modeling
Balanced_Acc_Test	If the distribution of two datasets in a dataset was not the same, this criterion was used to calculate the accuracy of the introduced method

### Assessable criteria

It was crucial to test and evaluate the results by a set of criteria to assess the performance of the proposed method. In general, the confusion matrix was used to evaluate the position and efficiency of the disease classification and diagnosis systems. The analysis of the confusion matrix in the classification and detection of flight delays led to the four modes of true positive, true negative, false positive, and false negative. Table 2 shows the position of the parameters in the confusion matrix.

The elements of the matrix were equal to:

In addition, the following criteria were used to evaluate the performance of the proposed method (Tables 3, 4).

### Results and discussion

Flight delays and the problem of predicting the amount of delay were divided into several factors, conditions, and data. According to a reliable study in this regard, flight delay predictions could be classified as:

1. Delays due to flight planning and scheduling;
2. Delays due to flight operation conditions at the airport;
3. Delays due to weather conditions;
4. Delays due to the terms and conditions of airline aviation operations and air traffic control;
5. Delays due to temporary conditions, such as the flight season or day;
6. Delays due to the flight conditions of the national flight network;
7. Delays due to the flight atmosphere

Since the type of delay in the present study included the numbers one (delays due to flight planning and scheduling), two (delays due to flight operation conditions at the airport), and six (delays due to the flight conditions of the national flight network), the amount of delay time slot was considered to be less than 15 min and 15–30 min based on the mentioned findings. A squawk radar is considered for the flight when the aircraft announces its readiness to fly based on the flight time specified in the flight schedule, and the flight will continue with the same squawk and flight sequence if it continues for 15 min. Otherwise, the squawk is canceled, and the flight must request a flight squawk from the country's air control center, which will change the flight schedule. On the other hand, if there is a delay of more than 15 min and less than 30 min, the flight can carry on with the same schedule and a new squawk. In case of a delay of more than 30 min, the

**Table 5** MSE and RMSE criteria

Criterion/airport	ALT	ORD	DFW	DEN	JFK	SFO	CLT	LAS	PHX
MSE (Delay 15)	1.68	1.78	1.62	1.52	1.63	9.73	3.04	5.21	2.68
MES (Delay 30)	0.8	1.38	1.4	0.97	1.32	6.55	4.98	4.25	3.4
RMSE (Delay 15)	1.84	1.33	1.28	1.23	1.27	3.11	1.74	2.28	1.63
RMES (Delay 30)	1.82	1.17	1.18	0.98	1.14	2.56	2.23	2.062	1.17

flight needs to send a flight delay message to the national air traffic network or set and send a new flight schedule.

**Calculate MSE and RMSE**

MSE is a statistical tool applied to determine predictive accuracy of a model. Table 5 shows the root-mean-square error (RMSE) of the desired airports. The parameter is mostly used to estimate the difference between the predicted values by a model and the observed values [1, 53]. The accuracy of the proposed model would be higher when the MSE per each specific mother was lower than the other model. The criteria considered in the proposed method for two delays of 15 and 30 min and 10 airports are presented in Table 5.

In other words, the higher predictive accuracy of a model leads to the lower MSE. The RMSE criteria in 30-min delays of LORD, PHX, and JFK airports had a lower percentage compared to the other airports, which was mainly due to the need for fewer traffic data compared to other airports, especially at the PHX Airport. In the case of the PHX Airport, the amount of air traffic data did not exceed the threshold value, while the traffic data for the other airports exceeded the threshold value [54].

**Accuracy**

The main purpose of this study is to increase accuracy. To better evaluate the proposed method, different situations have been considered based on these three scenarios:

- First scenario: 80% of data for learning and the remaining 20% for testing
- Second scenario: 60% of data for learning and the remaining 40% for testing
- Third scenario: 70% of data for learning and the remaining 30% for testing

Also by viewing all first scenarios you can say that the Third scenario had a higher accuracy (Table 6).

According to the findings, the LSTM-ELM hybrid method could detect the delay with the accuracy of 96.27. Accuracy varied at different time intervals of 15 and 30 min at various airports. According to the obtained results, the accuracy of the 30-min delays was higher at the ATL Airport. Nevertheless, the accuracy was acceptable in the other airports as well. The other criteria for delay in the other airports are compared in Table 7.

The first evaluated criterion was accuracy. As is observed, a 30-min delay had a higher accuracy percentage, a reason for which is that the delay has been obtained

**Table 6** Accuracy criteria for 15- and 30-min delays

Scenario	Delay/airport	ATL	LAX	ORD	DFW	DEN	JFK	SFO	CLT	LAS	PHX	Average
Scenario 1 80%-train 20%-test	Delay 15	98.46	88.34	88.34	98.22	98.64	98.48	90.27	96.96	94.24	97.32	94.92
	Delay 30	99.2	92.41	98.62	98.6	98.03	98.68	93.45	93.45	95.75	96.6	96.27
Scenario 2 60%-train 40%-test	Delay 15	49.99	49.98	98.48	98.36	97.83	97.28	97.28	93.51	49.92	86	79.07
	Delay 30	49.99	61.97	98.42	98.81	49.98	50	93.49	50	79.71	5.96	72.84
Scenario 3 70%-train 30%-test	Delay 15	98.71	50.34	98.28	98.64	49.99	98.31	92.22	94.99	92.74	0.8	85.42
	Delay 30	98.52	90.95	49.99	98.9	98.29	99.37	75.89	95.64	95.87	96.6	90

**Table 7** Comparison of evaluated criteria for 15- and a 30-min delays

ROW	AirPort	Delay	Recall	Specify	F_Mesure	Balanced_Acc	MCC
1	ATL	Delay 15	99.12	98.81	98.47	98.46	96.94
		Delay 30	99.7	98.69	99.2	99.2	98.4
2	LAX	Delay 15	99.78	77.89	89.44	88.34	78.4
		Delay 30	96.46	98.22	92.9	85.48	99.35
3	ORD	Delay 15	98.89	97.56	98.24	98.22	96.46
		Delay 30	99.35	97.88	98.63	98.62	97.24
4	DFW	Delay 15	98.84	98.44	98.64	98.64	97.28
		Delay 30	98.76	98.44	98.6	98.6	97.2
5	DEN	Delay 15	98.95	98.01	98.49	98.48	96.96
		Delay 30	99.36	98.71	99.04	99.03	98.07
6	JFK	Delay 15	99.28	97.55	98.43	98.42	96.85
		Delay 30	99.5	97.87	98.69	98.68	97.38
7	SFO	Delay 15	99.52	81.03	91.1	90.27	81.96
		Delay 30	99.6	87.3	93.83	93.45	7.56
8	CLT	Delay 15	99.18	94.73	97.02	96.96	94
		Delay 30	99.6	90.4	95.24	95.02	90.43
9	LAS	Delay 15	99.16	89.32	94.51	94.24	88.92
		Delay 30	99.52	91.98	95.9	95.75	91.76
10	PHX	Delay 15	99.36	95.28	97.37	97.32	94.72
		Delay 30	99.44	93.76	94.1	96.6	93.35

**Table 8** Comparison of evaluated criteria

Delay/airport	ATL	LAX	ORD	DFW	DEN	JFK	SFO	CLT	LAS	PHX
Accuracy [1]	93.45	85.19	90.49	89.65	87.52	85.64	88.83	93.25	85.76	9.88
Accuracy [54]	90.95	86.96	85.61	89.31	89.62	86.51	87.52	91.08	91.81	71.34
Accuracy proposed	98.46	88.34	98.22	98.48	98.48	98.42	90.27	96.96	94.24	97.32

and calculated due to flight operations in the TMA control space in estimating 30-min delays, which adds to the previous delays and could no longer be estimated.

Moreover, in delays of 30 min and more, the recorded information is more accurate since the order of flight arrival and departure numbers changes according to the order intended for the flight with the airport control mechanism and it is necessary to send a flight delay message or a flight plan update.

The second criteria evaluated in Table 7 was recall, which had a better percentage of 15-min delays at the LAX Airport. Some of the advantages of the data of these two airports included less noise and proximity to each other. This airport has the largest number of flights compared to the nearby airports, as well as a higher operating volume than other airports. The amount of system recall in the obtained estimate leads to the detection and reduction of human errors, operating systems, aviation accidents, and operational and airport costs. In addition, the three criteria of accuracy balance, MCC, and F-measure had better performance in 30-min delays. The use of the BiLSTM algorithm and improvement of the ELM parameter had a properly generalized 30-min delay. In addition, the improvement of the ELM in the training and testing phase will increase accuracy and precision compared to other airports. Therefore, it could be concluded

that the effect of the delay was properly modeled using the proposed method. In general, the improved ELM algorithm is faster, more accurate, and more generalizable in classification compared to other algorithms.

According to Table 7, the cause of 40% of the delays has been recorded in the airports, the most important of which was air time, followed by delayed arrival. Each of the delay factors alone could record several arrival delays at the subsequent airports, except for the arrival delay factor. Therefore, a significant part of the delay factors was related to delayed arrivals, which will be resolved when airlines have the required time for retrieving and returning to the flight schedule. At present, the cause of delays of less than 15 min and departure delays is not recorded at most airports. However, the recording information for the delays between 15 and 30 min is more thorough, which leads to higher accuracy and precision (Table 8).

#### ***Comparison of the proposed method with conducted research***

Improving the accuracy and precision of the ATM is a basic method in ATM research. Several ATM approaches have been provided on an ATC level. The accuracy of the proposed approach to traffic was low and did not respond to heavy traffic.

In the present study, an LST-ELM hybrid model was applied to improve the accuracy of the proposed method. The comparison of the proposed approach for the 30-min delay and [1] and [54] studies is shown in Table 8.

According to the obtained results, the proposed method had a more appropriate performance improvement as opposed to the comparable references due to the reconstruction of nonlinear time series and valid predictions. The obtained results also indicated that the proposed method could manage a complex nonlinear time series. Therefore, the use of the BiLSTM algorithm requires fewer hidden layers due to its greater learning capability and improving of the ELM network, which could enhance accuracy in an air traffic delay. Unlike other algorithms (e.g., BP), using the ELM algorithm needs no hidden layers, and its parameters are selected randomly. The goal of this algorithm is achieving the lowest training error and the smallest output soft weight. Furthermore, the improvement of this algorithm leads to the avoidance of the local minimum, and BiLSTM could be used to solve the long-term dependency problem. Together, these two algorithms improve accuracy more effectively compared to other methods.

#### **Conclusion**

The improved accuracy in ATM management problems is proposed in this paper. ATM includes all the necessary activates for the safe and useful management of the National Aviation System which is one the most challenging problems in our country's airports right now. In this paper, the dual-sided LSTM algorithm is used to improve the 15 and 30-min delays' accuracy. Also, this algorithm is used to improve the ELM algorithm's parameters. The data set used in this paper is taken from Kaggle and is a simulation used by MATLAB. The results show a higher accuracy improve the rate in comparison to other paper and also show that the RMSE parameter in 30-min delays has a lower percentage in ORD, PHX, and JFK airports in comparison to the other airports.

In further studies, to increase the ATM accuracy, other LSTM models like Casc-LSTM and Ens2-LSTM can be used alongside the ELM algorithm. One-way and two-way Lstm council can also be used along with other algorithms.

## Appendix

Automatic dependent surveillance-broadcast (ADS-B): is a surveillance technology in which an aircraft determines its position via satellite navigation and periodically broadcasts it, enabling it to be tracked.

Air traffic management (ATM): is an aviation term encompassing all systems that assist aircraft to depart from an aerodrome, transit airspace, and land at a destination aerodrome, including Air Traffic Services (ATS), Airspace Management (ASM), and Air Traffic Flow and Capacity Management (ATFCM).

Air Traffic Service (ATS): is a service which regulates and assists aircraft in real-time to ensure their safe operations. In particular, ATS is to:

- prevent collisions between aircraft; provide advice of the safe and efficient conduct of flights;
- conduct and maintain an orderly flow of air traffic;
- notify concerned organizations of and assist in search and rescue operations.

Bidirectional LSTMs: are an extension of traditional LSTMs that can improve model performance on sequence classification problems. In problems where all time steps of the input sequence are available, bidirectional LSTMs train two instead of one LSTMs on the input sequence.

Elapsed flying time: Actual time an airplane spends in the air, as opposed to time spent taxiing to and from the gate and during stopovers.

Extreme learning machines (ELM): are feed-forward neural networks for classification, regression, clustering, sparse approximation, compression and feature learning with a single layer or multiple layers of hidden nodes, where the parameters of hidden nodes (not just the weights connecting inputs to hidden nodes) need not be tuned.

Long short-term memory (LSTM): is an artificial recurrent neural network (RNN) architecture used in the field of deep learning. Unlike standard feed-forward neural networks, LSTM has feedback connections.

Terminal control area (TCA or TMA): A terminal control area (TMA, or TCA in the U.S. and Canada), also known as a terminal manoeuvring area (TMA) in Europe, is an aviation term to describe a designated area of controlled airspace surrounding a major airport where there is a high volume of traffic.

## Abbreviations

ADS-B: Automatic dependent surveillance-broadcast; ATC: Air traffic control; ATFM: Air traffic flow management; ATM: Air traffic management; ATS: Aircraft traffic service; B-LSTM: Bidirectional long short-term memory; DBN: Deep belief networks; ELM: Extreme learning machine; ETA: Estimated time of arrival; ETD: Estimated time of departure; LSTM: Long short-term memory; MSE: Mean squared error; NAS: National air system; RMSE: Root mean square error; RNN: Recurrent neural network; TCA: Traffic control area; TMA: Terminal manoeuvring area.

**Acknowledgements**

The authors are thankful to anonymous reviewers for their valuable comments and suggestions that helped improving the quality of the paper.

**Authors' contributions**

Not applicable. All authors read and approved the final manuscript.

**Funding**

Not applicable.

**Availability of data and materials**

Datasets that have been used for experiments in this paper are available at: kaggle.com.

**Declarations****Ethics approval and consent to participate**

Not applicable.

**Consent for publication**

Not applicable.

**Competing interests**

The authors have no conflict of interest.

**Author details**

<sup>1</sup> Department of Computer Engineering, Neyshobur Branch, Islamic Azad University, Neyshabur, Iran. <sup>2</sup> Department of Computer Engineering, Mashhad Branch, Islamic Azad University, Mashhad, Iran.

Received: 16 November 2020 Accepted: 13 March 2021

Published online: 01 April 2021

**References**

1. Kim YJ. A deep learning and parallel simulation methodology for air traffic management. Atlanta: Georgia Institute of Technology; 2017.
2. Sternberg A, et al. A review on flight delay prediction. Published 2017 in Arxiv, CEFET/RJ, Rio de Janeiro, Brazil. 2017.
3. Kistan T, et al. An evolutionary outlook of air traffic flow management techniques. *Prog Aerosp Sci*. 2017;88:15–42.
4. Henriques R, Feiteira I. Predictive modelling: flight delays and associated factors, Hartsfield-Jackson Atlanta International Airport. *Procedia Comput Sci*. 2018;138:638–45.
5. Riahi V, et al. Constraint guided search for aircraft sequencing. *Expert Syst Appl*. 2019;118:440–58.
6. Lieder A, Stolletz R. Scheduling aircraft take-offs and landings on interdependent and heterogeneous runways. *Transp Res Part E Logist Transp Rev*. 2016;88:167–88.
7. Oza S, et al. Flight delay prediction system using weighted multiple linear regression. *Int J Eng Comput Sci*. 2015;4(05).
8. Samà M, et al. Air traffic optimization models for aircraft delay and travel time minimization in terminal control areas. *Public Transp*. 2015;7(3):321–37.
9. Chen J, et al. Air traffic flow management under uncertainty using chance-constrained optimization. *Transp Res Part B Methodol*. 2017;102:124–41.
10. Takeichi N, et al. Prediction of delay due to air traffic control by machine learning. In: AIAA modeling and simulation technologies conference. 2017.
11. Messaoud MB, et al. Detailed mathematical programming formulations for the aircraft landing problem on a single and multiple runway configurations. *Procedia Comput Sci*. 2018;126:345–54.
12. Ivanov N, et al. Air traffic flow management slot allocation to minimize propagated delay and improve airport slot adherence. *Transp Res Part A Policy Pract*. 2017;95:183–97.
13. Faye A. A quadratic time algorithm for computing the optimal landing times of a fixed sequence of planes. *Eur J Oper Res*. 2018;270(3):1148–57.
14. Santos BF, et al. Airline delay management problem with airport capacity constraints and priority decisions. *J Air Transp Manag*. 2017;63:34–44.
15. Bertsimas D, Frankovich M. Unified optimization of traffic flows through airports. *Transp Sci*. 2015;50(1):77–93.
16. Aljubairy A, et al. Real-time investigation of flight delays based on the internet of things data. In: International conference on advanced data mining and applications. Springer. 2016.
17. Furini F, et al. Improved rolling horizon approaches to the aircraft sequencing problem. *J Sched*. 2015;18(5):435–47.
18. Santos PLCT, et al. A methodology used for the development of an Air Traffic Management functional system architecture. *Reliab Eng Syst Saf*. 2017;165:445–57.
19. Kwasiborska A. Sequencing landing aircraft process to minimize schedule length. *Transp Res Procedia*. 2017;28:111–6.
20. Hrastovec M, Solina F. Prediction of aircraft performances based on data collected by air traffic control centers. *Transp Res Part C Emerg Technol*. 2016;73:167–82.
21. Busquets JG, et al. Application of data mining to forecast air traffic: a 3-stage model using discrete choice modeling. 2016.



22. Alexander DW, Merkert R. Challenges to domestic air freight in Australia: evaluating air traffic markets with gravity modelling. *J Air Transp Manag.* 2017;61:41.
23. Simaiakis I, Balakrishnan H. A queuing model of the airport departure process. *Transp Sci.* 2015;50(1):94–109.
24. Ghoniem A, et al. Enhanced models for a mixed arrival-departure aircraft sequencing problem. *INFORMS J Comput.* 2014;26(3):514–30.
25. Baomar H, Bentley PJ. Autonomous navigation and landing of airliners using artificial neural networks and learning by imitation. In: 2017 IEEE symposium series on computational intelligence (SSCI). 2017.
26. Wang Z, et al. A hybrid machine learning model for short-term estimated time of arrival prediction in terminal manoeuvring area. *Transp Res Part C Emerg Technol.* 2018;95:280–94.
27. Choi S, et al. Prediction of weather-induced airline delays based on machine learning algorithms. In: Digital avionics systems conference (DASC), 2016 IEEE/AIAA 35th, IEEE. 2016.
28. Pagels DA. Aviation Data Mining. *Sch Horiz Univ Minnesota Morris Undergrad J.* 2015;2(1):3.
29. Tang X, et al. A flight profile clustering method combining twed with K-means algorithm for 4D trajectory prediction. In: Integrated communication, navigation, and surveillance conference (ICNS), 2015, IEEE. 2015.
30. Wu Y, et al. A sequencing model for a team of aircraft landing on the carrier. *Aerosp Sci Technol.* 2016;54:72–87.
31. Caraka RE, Lee Y, Chen RC, Toharudin T. Using hierarchical likelihood towards support vector machine: theory and its application. *IEEE Access.* 2020;8:194795–807.
32. Suhermi N, Prastyo DD, Ali B. Roll motion prediction using a hybrid deep learning and ARIMA model. *Procedia Comput Sci.* 2018;144:251–8.
33. Alligier R, Gianazza D. Learning aircraft operational factors to improve aircraft climb prediction: a large scale multi-airport study. *Transp Res Part C Emerg Technol.* 2018;96:72.
34. Zhang M, et al. Analysis of flight conflicts in the Chinese air route network. *Chaos Solitons Fractals.* 2018;112:97–102.
35. Bongiorno C, et al. Statistical characterization of deviations from planned flight trajectories in air traffic management. *J Air Transp Manag.* 2017;58:152–63.
36. Gopalakrishnan K, Balakrishnan H. A comparative analysis of models for predicting delays in air traffic networks, ATM Seminar. 2017.
37. Kuhn N, Jamadagni N. Application of machine learning algorithms to predict flight arrival delays. 2017.
38. Lovato AV, et al. A fuzzy modeling approach to optimize control and decision making in conflict management in air traffic control. *Comput Ind Eng.* 2018;115:167–89.
39. El Hatri C, Boumhidi J. Fuzzy deep learning based urban traffic incident detection. *Cogn Syst Res.* 2018;50:206–13.
40. Sarabakha A, et al. Novel Levenberg–Marquardt based learning algorithm for unmanned aerial vehicles. *Inf Sci.* 2017;417:361–80.
41. Tong C, et al. A novel deep learning method for aircraft landing speed prediction based on cloud-based sensor data. *Future Gen Comput Syst.* 2018;88:552–8.
42. Asadi F, Richards A. Ad hoc distributed model predictive control of air traffic management. *IFAC-PapersOnLine.* 2015;48(25):68–73.
43. Lehouillier T, et al. Measuring the interactions between air traffic control and flow management using a simulation-based framework. *Comput Ind Eng.* 2016;99:269–79.
44. Su Y, Kaiquan C. A multi-objective multi-memetic algorithm for network-wide conflict-free 4D flight trajectories planning. *Chin J Aeronaut.* 2017;30(3):1161–73.
45. Gardi A, et al. Multi-objective optimisation of aircraft flight trajectories in the ATM and avionics context. *Prog Aerosp Sci.* 2016;83:1–36.
46. Zhang Y, et al. Sector-based distributed scheduling strategy in air traffic flow management. *IFAC-PapersOnLine.* 2016;49(3):365–70.
47. Polvara R, et al. Toward end-to-end control for UAV autonomous landing via deep reinforcement learning. In: 2018 International conference on unmanned aircraft systems (ICUAS), IEEE. 2018.
48. Ghomi SF, Forghani K. Airline PASSENGER forecasting using neural networks and Box-Jenkins. In: Industrial engineering (ICIE), 2016 12th international conference on, IEEE. 2016.
49. Ni X, et al. Civil aviation safety evaluation based on deep belief network and principal component analysis. *Saf Sci.* 2019;112:90–5.
50. Yang H-F, et al. Optimized structure of the traffic flow forecasting model with a deep learning approach. *IEEE Trans Neural Netw Learn Syst.* 2017;28(10):2371–81.
51. Tong C, et al. An innovative deep architecture for aircraft hard landing prediction based on time-series sensor data. *Appl Soft Comput.* 2018;73:344–9.
52. Venkatesh V, et al. Iterative machine and deep learning approach for aviation delay prediction. 978-1-5386-3004-4/17/\$31.00 ©2017 IEEE. 2017.
53. Choi S. A multi-level predictive methodology for terminal area air traffic flow. Georgia Institute of Technology. 2019.
54. Kim YJ, Choi S, et al. A deep learning approach to flight delay prediction. 978-1-5090-2523-7/16/\$31.00 ©2016 IEEE. 2016.
55. Rong F, et al. The prediction of flight delays based the analysis of random flight points. In: Control conference (CCC), 2015 34th Chinese, IEEE. 2015.
56. Hong liu, Yi lin, , et al. Research on the air traffic flow prediction using a deep learning approach. *IEEE ACCESS Digit Object Identif.* 2019. <https://doi.org/10.1109/ACCESS.2019.2945821>.
57. Naessens H, et al. Predicting flight routes with a Deep Neural Network in the operational Air Traffic Flow and Capacity Management system. 17 December 2017 EUROCONTROL Maastricht Upper Area Control Centre Horsterweg 11, NL-6199 AC Maastricht Airport. 2019.
58. Lambelho M, et al. Assessing strategic flight schedules at an airport using machine learning-based flight delay and cancellation predictions. *J Air Transp Manag.* 2020;82:101737.
59. Lin Y, et al. Deep learning based short-term air traffic flow prediction considering temporal–spatial correlation. *Aerosp Sci Technol.* 2019. m5G; v1.256; Prn:17/04/2019; 12:58] P.1(1–9)

60. Zhang H, et al. Airport delay prediction based on spatiotemporal analysis and bi-LSTM sequence learning. 978-1-7281-4094-0/19/\$31.00 ©2019 IEEE, 2019.
61. Toharudin T, Pontoh RS, Caraka RE, Zahroh S, Lee Y, Chen RC. Employing long short-term memory and Facebook prophet model in air temperature forecasting. *Commun Stat Simul Comput*. 2020. <https://doi.org/10.1080/03610918.2020.1854302>.
62. Caraka RE, Chen RC, Supatmanto BD, Tahmid M, Toharudin T. Employing moving average long short term memory for predicting rainfall. In: 2019 international conference on technologies and applications of artificial intelligence (TAAI). IEEE. 2019, November. pp. 1–5.
63. Reitmann S, Nachtigall K. Applying bidirectional long short-term memories (BLSTM) to performance data in air traffic management for system identification. Springer International Publishing AG 2017 A. Lintas et al. editors. ICANN 2017, Part II, LNCS 10614, pp. 528–536, 2017. [https://doi.org/10.1007/978-3-319-68612-7\\_60](https://doi.org/10.1007/978-3-319-68612-7_60)

### **Publisher's Note**

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

**Submit your manuscript to a SpringerOpen<sup>®</sup> journal and benefit from:**

- ▶ Convenient online submission
- ▶ Rigorous peer review
- ▶ Open access: articles freely available online
- ▶ High visibility within the field
- ▶ Retaining the copyright to your article

---

Submit your next manuscript at ▶ [springeropen.com](https://www.springeropen.com)

---