# Prediction of Air Flight Cancellation during COVID-19 using Deep Learning Methods

[1]Shawni Dutta and [2]Prof. Samir Kumar Bandyopadhyay

[1]Lecturer, Department of Computer Science, The Bhawanipur Education Society College, Kolkata, India

[2]Academic Advisor, The Bhawanipur Education Society College, Kolkata, India

## Abstract

Air traffic is vulnerable to external factors, such as oil crises, natural disasters, economic recessions and disease outbreaks due to COVID-19. This reason seems to have a more severe and more rapid impact on air traffic numbers as sudden increases in flight cancellations, aircraft groundings and travel bans. Various Airways loose revenues and it is difficult for them to sustain for a long period. This problem as been facing the entire world. The reductions in passenger numbers are significant. It is due to flights being cancelled or planes flying empty between airports. It is in turn massively reducing revenues for airlines and forced many airlines to lay off employees or declare bankruptcy. Airways also have to attempt refunding cancelled trips in order to diminish their losses. The airliner manufacturers and airport operators have also laid off employees. According to some commentators,this crisis is the worst ever encountered in the history of the aviation industry.

Aircraft cancellation prediction is accomplished by utilising deep learning framework. In this framework, two dissimilar recurrent neural networks are assembled as a single entity while inferring the prediction results. Long-short term memory (LSTM) and Gated Recurrent Unit (GRU) are employed to design the proposed predictive model. This predictive model is compared against traditional neural network based Multi-layer perceptron model. Experimental results indicated an accuracy of 98.7% by the proposed model.

**Key-Words: COVID-19; Flight Cancellation; Neural network;GRU; LSTM; RNN**

## 1. Introduction

Flight services are highly affected due to adverse weather condition that may lead to flight cancellation. Security issues, mechanical issues, air traffic restrictions, bird strikes, unavailability of aircrafts, deficiency of crew members are also responsible for flight cancellation. A study revealed that the reporting carriers cancelled 1.6% of their scheduled domestic flights in December 2016, an improvement over the 1.7% cancellation rate posted

in December 2015, but up from the 0.3 % rate in November 2016. Passengers are affected by delays and cancellations. The waiting in the airport due to announcement of delay of particular flight cause stress among passengers. This disrupts the purpose of air travel for rapid journey from one place to another place [1].

The Governments around the world decided to temporarily introduce travel restrictions due to COVID-19 pandemic. As a result, that the airlines were forced to suspend their flights. The passengers received notifications about cancellation or information that the cancellation of their flights may be delayed. Airlines are now introduced around the world have introduced new regulations for ticket changes and returns in the current situation. These regulations are applied only to flights whose cancellation has been confirmed.

Passengers can have less stress if the airline has informed it earlier that the flight has been cancelled. It is better if an e-mail was sent to the passenger earlier so that he/she will not move to the airport from house, if possible, and confirms the cancellation of ticket. Passengers are entitled to a refund from the airline for the unused ticket. The possible refund option can be known from airlines. Airline authority sent a message with the refund options allowed by the airline. The authority also informed the passenger that the cancellation of flight is due to COVID-19.

This paper attempts to predict flight cancellation by involving data mining techniques. Data mining techniques are often useful in finding interesting patterns from the existing databases by applying sophisticated algorithms. The acquired patterns make knowledge base and thus helpin taking informed decisions [2]. This study practices predictive modelling tasks of data mining techniques for achieving flight cancellation detection. In this context, a deep learning (DL) [3] based mechanism is exemplified for fulfilling the aforementioned target of this research. Recurrent Neural Network (RNN) is a popular DL technique that accomplishes predictive modelling tasks. LSTM and GRU [4] are two popular RNN that are integrated under a single platform along with certain hyper-parameter tuning.

The objective of this research can be summarised as follows-

1. Extract flight transportation data and apply necessary pre-processing techniques for obtaining cleaned dataset.
2. Use DL technique for aircraft cancellation prediction.

3. A combined RNN based method is exemplified for this purpose. Two variations of RNN such as LSTM and GRU are combined for designing the proposed classification model. This model is compared against benchmark classifier model such as Multi-layer perceptron model.

As contrast to major researches such as [5-10], instead of predicting flight delay scenarios, flight cancellation detection is favoured in this paper. Flight departure delay may appear be to one of the severe reasons for flight cancellations. Hence, it is necessary to consider the impact of delays on flight cancellation event. A computer aided classification can help to predict aircraft cancellation which in turn can save the resources and optimise the passenger's anxiety.

## 2. Related Works

Many researches had been carried out while retrieving flight delay predictions. For analysing arrival delay of flights a comparative study had been carried out in [5] for developing classifier model. The classifier models include random forest, Support Vector Machine (SVM), Gradient Boosting Classifier (GBC) and k-nearest neighbour algorithm. Comparative results show that the gradient boosting classifier model attains the best predictive arrival delay performance of 79.7% of total scheduled American Airlines' flights. Another study in [6] applied random forest network-based air traffic delay prediction models considering both temporal and network delay states. Application of deep learning techniques was presented in [7] for analysing the patterns in air traffic delays. Long Short-Term Memory RNN architecture was utilised for building predictive model. Tu et. al. [8] presented a model that considers both seasonal trend and daily propagation patterns. Applying statistical methods for analysing long-term and short-term patterns in air traffic delays, departure delays had been estimated. While investigating propagation delays among airports, Bayesian network had been exemplified in [9]. For achieving the same objective, i.e., forecasting possible delays, multilevel input layer artificial neural network (ANN) was utilised in [10].

Another event flight cancellation has been carried out by [11] using support vector machine (SVM), decision tree (DT), Naïve Bayes, Logistics Regression (LR) methods. Their prediction performances were compared and an accuracy of 90% was reached by SVM and LR. Best of our knowledge, flight cancellation prediction has not been much studied yet. This field can still be explored and hence in this paper focuses on carrying out the prediction.

## 3. Background

### 3.1 Neural Network and Deep Learning

Deep Learning (DL), is a subfield of Machine Learning (ML), automates and develops the machines which is the goal eventually exhibited by Artificial Intelligence (AI). By assembling more than two layers, DL provides a multi-layered hierarchical data representation typically in the form of a neural network. It is beneficial since it does not include the manual feature engineering task to be performed due to its self-adaptive nature. It establishes the involvement of neural network in order to accompany complex problem solving approach. A large number of processing elements (nodes) are present in neural network like neurons in human brain for acquiring best problem solving tactic [2].

### 3.2 Hyper-parameters used in neural network training

Some Pre-stage fine-tuning of hyper-parameters is necessary to perform before training this neural network. It contains number of layers, number of nodes, learning rate, epoch size, batch size and drop-out rate. These values should be adjusted to help the network to learn successfully. Activation function is one of the necessary tasks which can maximize the training procedure. These functions allow neural networks to learn non-linear relationship among data and to produce meaningful output signal. Sigmoid activation function may be used for activating output nodes for predicting binary class probabilities. The activation function accepts the input data and transforms it in the range of 0 to 1 and it is shown in equation (1) [12]. Tangent hyperbolic (tanh) is also non-linear activation function and it is a smoother and zero-centered function [12]. The function range in between -1 to 1 and the output of the Tanh function is given in equation (2).

$$f(x) = 1/(1 + \exp^{-x}) \quad (1)$$

$$f(x) = (e^{x} - e^{-x})/(e^{x} + e^{-x}) \quad (2)$$

For eliminating over-fitting problem in neural networks the dropout technique is employed. In the training process it randomly detaches units along with incoming and outgoing connections from the neural network. Neural network acquires benchmark results in supervised classification tasks by using dropout [13].

Hyper-parameters such as epoch and batch size are also used in neural network training. These hyper-parameters receive integer values which need to be chosen wisely to make best

use of the model's performance. The size of the epoch is defined to be number of passes to complete through training dataset. The dataset is passed forward and backward through the neural network exactly one time within each epoch. During passing the entire dataset into the algorithm, it must be partitioned into fixed size of batches. The size of the batch keeps track of number of processed instances before the model updates its internal parameters. It needs to be ensured that batch size should not be too small or too large. If the size of the batch is too small then it will present high variance. It means that it does not represent the entire dataset. On the other hand, large batch size may not fit in memory to compute samples used for training and may lead to over-fitting problem [14].

The use of optimizer is mandatory in order to stack multiple Recurrent Neural Network (RNN) layers under a single framework,. Adam is is computationally efficient with optimised memory requirement and also easy to implement. The proposed method can be applied to optimize for first-order gradient-based optimization of stochastic objective functions, based on adaptive estimates of lower-order moments. It is quite well accepted due to its applicability on non-stationary objectives and problems with very noisy and/or sparse gradients [15].

### 3.3 Recurrent Neural Network, LSTM and GRU

RNN is constructed using multiple neural networks which are specialized for analyzing sequential data. In this network, the output of previous step is fed into current step input. For example, the output obtained during step Si affects the parameters of step Si+1. Hence, it is quite vibrant that RNN accepts two types of input- one is present input and other one is previous output for acquiring the final output. Loops present in RNN allow the signals to travel both forward and backward. However, RNN suffers from the problem of vanishing gradient problem [16].

To resolve this mentioned problem, variants of RNN Long short-term memory (LSTM) and Gated Recurrent Units (GRU) are explored. This variant contains gates which are neural networks that control the flow of information through the sequence chain. To mitigate short-term memory, methods like LSTM and GRU introduce the concept of gates. LSTM neural network is a kind of RNN that implements context based prediction which is not considered in traditional RNN. In other words, LSTM is capable to eliminate the problem of vanishing gradient by training RNN. LSTM has a good potential to regulate gradient flow as well as

better preservation of long-range dependencies. Every cell in LSTM is comprised of gates that determine when to remember input, when to remember or forget the value and when it should output the value. Depending on the performance, there are variants in gates such as input gate, output gate and forget gate. The input gate blocks a value from entering into next layer when a value close to zero is generated by this gate. This input gate simply eliminates the value from the net input. Forget gate remembers value until greater value than zero is generated by forget gate. When value closes to zero value is produced, the block effectively forgets the value to be remembered. The output gate determines when the unit should output the value in its memory [17].

Equation 3 to equation 8 describes$x_t = (x_1, \ldots, x_T)$ be an input sequence and output sequence $y_t = (y_1, \ldots, y_T)$. The weight matrices is W, b is the bias vector, σ states the sigmoid function, and i, f ,o, c are the input gate, forget gate, output gate, and cell activation vectors, respectively. denotes The weight's matrix is denoted by $W_{ix}$. $W_{ic}$ , $W_{fc}$ , $W_{oc}$ denote diagonal weight matrices for peephole connections, In addition, ⊙ and Ø indicate element-wise multiplication and softmax activation function for the LSTM.

$$i_t = \sigma\,[(W_{ix} \times x_t) + (W_{ir} \times r_{t\ 1}) + (W_{ic} \times c_{t\ 1}) + b_i] \quad (3)$$

$$f_t = \sigma\,[\,(W_{fx} \times x_t) + (W_{fr} \times r_{t\ 1}) + (W_{fc} \times c_{t\ 1}) + b_i\,] \quad (4)$$

$$c_t = f_t \odot c_{t\ 1} + i_t \odot \tanh\,[(W_{cx} \times x_t) + (W_{cr} \times r_{t\ 1}) + b_c] \quad (5)$$

$$o_t = \sigma\,[(W_{ox} \times xt) + (W_{or} \times r_{t\ 1}) + (W_{oc} \times ct) + b_o] \quad (6)$$

$$r_t = o_t \odot \tanh\,(c_t) \quad (7)$$

$$y_t = Ø(W_{yr} \times rt + b_y) \quad (8)$$

GRUis a gating mechanism in RNN similar to a long short-term memory (LSTM) unit. It is used without an output gate. GRU is considered a variation of the LSTM because both have a similar design and produce equal results in some cases. In GRU the update gate controls information that flows into memory and the reset gate controls the information that flows out of memory. The two vectors decide which information will get passed on to the output. GRU can be trained to keep information from the past or remove information that is irrelevant to the prediction.

In sequence learning tasks and overcome the problems of vanishing and explosion of gradients, Gated Recurrent unit (GRU) networks perform well when it is learning long-term dependencies. GRU consists of two types of gates such as update and reset gate. Addition and elimination of information is decided by update gate. The use of reset gate identifies how much information to hold from past. GRU uses update gate and reset gate for solving the vanishing gradient problem of a standard RNN. These vectors decide what information should be passed to the output. Without washing it through time or remove information, vectors can be trained to keep information from long ago. It is irrelevant to the prediction [18].

Given $x_t = (x_1, \ldots, x_T)$ be an input sequence, W is the weight matrices σ states the sigmoid function for a GRU. At time t, the activation function of GRU is $h_j^t$ which is dependent on previous activation $h_{t-1}^j$ candidate activation function $h`_t^j$. This is formulated in equation (9). The update gate ($u_t^j$), and reset gate ($r_t^j$) can be formulated as equation (10) and equation (11) respectively.

$$h_j^t = (1 \quad u_t^j)h_{t-1}^j + u_t^j h'^j_t \qquad (9)$$

$$u_t^j = \sigma\,(W_u \,.\, [h_{t-1}^j \,, x_i]) \qquad (10)$$

$$r_t^j = \sigma(W_r \,.\, [h_{t-1}^j \,, x_i]) \qquad (11)$$

### 3.4 Model Evaluation

Evaluation metrics are taken into consideration while discriminating the performance of any model from other models. Accuracy and loss are required to calculate for any deep model. For each epoch, accuracy and loss are calculated during training efficiency assessment. A loss function (or cost function) [19] measures how much the model makes mistakes for each instance in the training set. In other words, the loss function acquires the probabilities of how much predicted values get varied from original value. Cross-entropy function can be used as loss function basically for binary classification problems. This function measures the performance of a classification model whose output is a probability value between 0 and 1 [20].

Using true positive (TP), true negative (TN), false positive (FP), false negative (FN), accuracy and f1-score metrics can be evaluated as equation (12) and (15) respectively. It is to be noted that, f1-score is a metric that relies on calculation of recall and precision which are formulated as equation (13) and (14) respectively [21].

Accuracy= TP+TN/(TP+FP+TN+TP) (12)

Recall= TP/(TP+FN) (13)

Precision= TP/(TP+FP) (14)

F1- Measure or F1-Score= 2* Recall * Precision / (Recall + Precision) (15)

Mean Squared Error (MSE) [21] is another evaluating metric that measures absolute differences between the prediction and actual observation of the test samples. MSE produces non-negative floating point value and a value close to 0.0 turns out to be the best one. It is formulated as equation (16).

MSE= $(\sum_{i=1}^{N}(Xi \quad Xi')^2 / N)$;where Xi is the actual value and Xi' is the predicted value

(16)

Cohen-Kappa Score [22] is also considered to be as an evaluating metric in this paper. This metric is a statistical measure that finds out inter-rate agreement for qualitative items for classification problem. It is formulated as equation (17).

$$\text{Cohen-Kappa Score} = (p_0 - p_e)/(1 - p_e) \quad (17)$$

where $p_o$ denotes relative observed agreement among raters and $p_e$ is the probability of agreement by chance.

Receiver Operating Characteristic (ROC) curve is used to visualize the relationship between true positive rate (alternatively known as recall) and false positive rate. The equations are shown in (18) and (19).

True positive rate (TPR)= TP/ (TP+FN) (18)

False positive rate (FPR)=FP/(FP+TN) (19)

Area under ROC Curve, often abbreviated as AUC, provides an aggregate measure of performance across all possible classification thresholds. AUC has values ranging from 0 to 1. A model that produces AUC as 1 can be regarded as best performing model whereas a model showing all wrong predictions will signify 0 as AUC [23].

## 4. Materials and Methods

### 4.1 Problem definition

This paper attempts to explore the predictive results for flight cancellation on a particular link for specific airport. Delay at time t may have impact on flight cancellation on time t+1. If the delay is quite larger, the tendency of flight cancellation will be successful. However, the aircraft cancellation for a particular flight also depends on distance between origin and destination, flight diversion tendency, flight timings. In this context, classifier model can be built where the output is binary prediction of whether a particular flight will be cancelled or not.

### 4.2 Data Source

To carry out the flight cancellation prediction, this research collects the data containing all the flights of USA in the month of January 2019 and January 2020. This open-source data under U.S. Govt. Works has been retrieved from kaggle data repository [24]. The dataset is consisting of 1191331 records with some missing values. The figure 1 shows the summary of missing values present in the dataset.

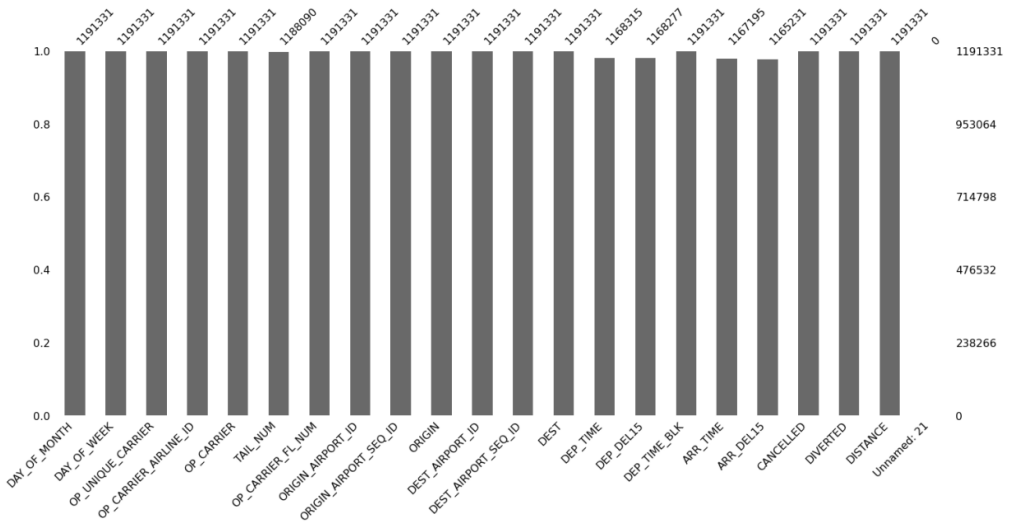


Figure 1: Missing values in dataset

As shown in figure 1, the attribute like 'Unnamed: 21' has no use. Similarly, the unique ID numbers present in this dataset are also eliminated from the dataset. Other attributes having missing values in the dataset is replaced by zero. Performing these steps will transform the dataset into cleaned one. This transformed dataset is bifurcated into training and testing set. The training and testing dataset is distinguished by the presence of attribute 'CANCELLED'. Since, prediction needs to be retrieved from testing dataset; the target attribute is eliminated from it. The classification process is carried out in such a way that it receives training for January 2019 and later prediction is retrieved for January 2020. In other words, flight detail of January 2019 is the training dataset whereas flight detail of January 2020 is the testing dataset for which prediction result is retrieved. Any classifier model learns from the training dataset by extracting hidden patterns and utilizes that knowledge while predicting unknown sample from testing dataset.

### 4.3 Methodology

The research methodology considered by this study focuses on utilising neural network based framework. For this reason, two models are presented. One model is stacked GRU-LSTM-RNN model which follows DL architecture and it is presented as proposed model by this paper. The other model is traditional neural network based MLP classifier model. The former model is compared with the baseline neural network classifier such as MLP.

#### 4.3.1 Stacked GRU-LSTM Model

The proposed stacked GRU-LSTM model considers aircraft cancellation event by implementing DL based framework. The predictive model consists of one GRU layer, one LSTM layer and four fully connected layers. The LSTM and GRU layers are followed by applying drop out layers that prevent the model from over-fitting. Hence, this model consists of total 8 layers. All these layers are stacked into a single model by using 'adam' optimizer and with a batch size of 64. Due to the enormous amount of data present in the dataset, an epoch size of 2 is favoured in this case. The model is trained for exactly two epochs and the testing data is used to evaluate the model. The detailed description of this model is provided in table 2. Description is provided in terms of number of units for each layer, dropout rate, amount of received parameters, activation function used.

Table 2: Implementation details of stacked GRU-LSTM model

| **Layer** | **Type** | **Units/Rate** | **Parameters** | **Activation function** |
|---|---|---|---|---|
| Layer 1 | GRU | 128 | 49920 | Sigmoid |
| Layer 2 | Dropout | 0.2 | 0 | None |
| Layer 3 | LSTM | 32 | 20608 | Sigmoid |
| Layer 4 | Dropout | 0.2 | 0 | None |
| Layer 5 | Dense | 8 | 264 | Tanh |
| Layer 6 | Dense | 4 | 36 | Tanh |
| Layer 7 | Dense | 2 | 10 | Tanh |
| Layer 8 | Dense | 1 | 3 | Sigmoid |
| **Epoch Size** | 2 | | | |
| **Batch size** | 64 | | | |
| **Optimizer** | Adam | | | |

### 4.3.2Multilayer Perceptron Model

Neural networks are designed to model the function of human brain in order to perceive complicated task. Multi-layer perceptron (MLP) classifier relies on neural network architecture for performing the task of classification. It follows a feed-forward neural network that accepts input features and maps them into appropriate output sets. Any MLP classifier comprises of three layers known as input layer, hidden layer and output layer. Hidden layers exist between the input layer and output layer. The number of hidden layers depends on problem-specific domain. When only one hidden layer exists in the architecture, it is often referred as Vanilla neural network. Introduction of several layers is determined by the requirement to increase the complexity of decision regions. The connections between perceptron in an MLP are forward and every perceptron is connected to all the perceptron in the next layer except the output layer that directly gives the result. MLP is advantageous since it can distinguish data those are linearly non-separable. MLP uses both linear as well as non-linear activation functions. Use of multiple layers along with non-linear activation functions makes MLP superior from linear perceptron. MLP follows supervised learning method that utilizes back propagation algorithm for training purpose [25].

This classifier model is built by considering several parameters such as number of hidden layers, size of hidden layers, optimizer used, activation function employed. These implementation details are summarised in table 3.

Table3: Hyper-parameters used for MLP classifier

| | |
|---|---|
| Activation Function | ReLu |
| Optimizer | Adam |
| Number of hidden layers | 3 |
| Size of hidden layers | 32,16,8 |

## 5. Experimental Results

Once the model is designed, the training process is executed through 2 epochs. The training procedure is evaluated against two metrics such as loss and accuracy exhibited by the model. The trade-off between loss and accuracy are shown in figure2. This shows the loss acquired by the model in the first epoch decreases as it reaches in the second epoch. Similarly, the training accuracy increases as the epoch proceeds. After the training process is completed, the testing process takes place. Table 4 gives summary of accuracy, f1-score, cohen-kappa score, MSE, Area under ROC curve exhibited by stacked GRU-LSTM model during the testing phase. Another implemented model such as MLP classifier is also summarised in table 4 in terms evaluation metrics. The comparative study signifies the superiority of stacked GRU-LSTM model over MLP classifier. Hence, the proposed model can be favoured as an accurate and robust one in the domain of flight cancellation prediction.

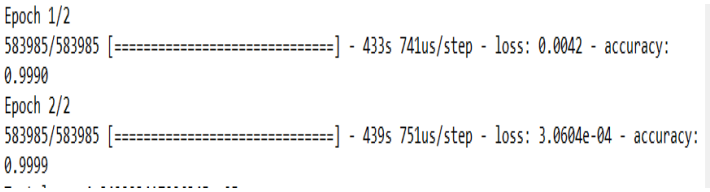


Figure2. Loss and Accuracy acquired for each epoch.

| | Accuracy | Cohen-kappa score | F1-Score | Area Under ROC Curve | MSE |
|---|---|---|---|---|---|
| Stacked GRU-LSTM Model | 98.7% | 0.978 | 0.988 | 0.98 | 0.017 |
| MLP Classifier | 97.3% | 0.967 | 0.972 | 0.965 | 0.021 |

Table4: Performance summary of Stacked GRU-LSTM and MLP classifier model.

## 6. Conclusions

For minimizing the substantial loss due to flight cancellation event, our research has been carried out to predict flight cancellation in advance. This study exemplifies the use of DL techniques for achieving the above mentioned purpose. The target of the proposed stacked GRU-LSTM predictive model is to provide insight regarding flight cancellations beforehand with promising accuracy and optimised error-rate. The designing of proposed model is accompanied by adjusting hyper-parameter tuning in order to maximize the predictive performance. While obtaining the prediction results, multiple interfering factors such as;the impact of flight delays, flight timings, distance between airports etc. are taken into consideration. The proposed model is capable toprovide prediction with an accuracy of 98.7% and MSE of 0.017.

## References

[1] 2016 Flight Cancellation, Mishandled Baggage, and Bumping Rates are Lowest in Decades. Retrieved from https://www.bts.dot.gov/newsroom/december-2016-airline-on-time-performance/#:~:text=The%20reporting%20carriers%20canceled%201.6,percent%20rate%20in%20November%202016. Accessed on 17 Aug 2020.

[2] Kurgan, L. A., &Musilek, P. (2006). A survey of knowledge discovery and data mining process models. *The Knowledge Engineering Review*, *21*(1), 1-24.

[3]Vargas, R., Mosavi, A., & Ruiz, R. (2017). Deep learning: a review. *Advances in Intelligent Systems and Computing*.

[4] Shrestha, A., & Mahmood, A. (2019). Review of Deep Learning Algorithms and Architectures. IEEE Access, 7, 53040-53065.

[5]Chakrabarty, N., Kundu, T., Dandapat, S., Sarkar, A., &Kole, D. K. (2019). Flight Arrival Delay Prediction Using Gradient Boosting Classifier. In *Emerging Technologies in Data Mining and Information Security* (pp. 651-659). Springer, Singapore.

[6]Rebollo, J.J., &Balakrishnan, H. (2014). Characterization and prediction of air traffic delays. Transportation Research Part C-emerging Technologies, 44, 231-241.

[7] Kim, Y., Choi, S., Briceno, S., &Mavris, D. (2016). A deep learning approach to flight delay prediction. 2016 IEEE/AIAA 35th Digital Avionics Systems Conference (DASC), 1-6.

[8]Tu, Y., Ball, M., &Jank, W. (2008). Estimating Flight Departure Delay Distributions - A Statistical Approach With Long-Term Trend and Short-Term Pattern. Journal of the American Statistical Association, 103, 112-125.

[9] Xu, N., Donohue, G., Laskey, K.B., & Chen, C. (2005). Estimation of Delay Propagation in the National Aviation System Using Bayesian Networks. Natural Computation, 2008. ICNC '08, Fourth International Conference, 4.

[10] Khanmohammadi, S., Tutun, S., &Kucuk, Y. (2016). A New Multilevel Input Layer Artificial Neural Network for Predicting Flight Delays at JFK Airport. Procedia Computer Science, 95, 237–244. doi:10.1016/j.procs.2016.09.321

[11] Yu, Y., Hai, M., & Li, H. (2019). A Classification Prediction Analysis of Flight Cancellation Based on Spark. ITQM.

[12] Nwankpa, C., Ijomah, W., Gachagan, A., & Marshall, S. (2018). Activation Functions: Comparison of trends in Practice and Research for Deep Learning. ArXiv, abs/1811.03378.

[13] Srivastava, N., Hinton, G.E., Krizhevsky, A., Sutskever, I., &Salakhutdinov, R. (2014). Dropout: a simple way to prevent neural networks from overfitting. J. Mach. Learn. Res., 15, 1929-1958.

[14] Radiuk, P. (2017). Impact of Training Set Batch Size on the Performance of Convolutional Neural Networks for Diverse Datasets. Information Technology and Management Science, 20, 20 - 24.

[15] Kingma, D.P., & Ba, J. (2015). Adam: A Method for Stochastic Optimization. CoRR, abs/1412.6980.

[16] Sherstinsky A. Fundamentals of Recurrent Neural Network (RNN) and Long Short-Term Memory (LSTM) network. Physical D: Nonlinear Phenomena 2020 Mar;404:132306.

[17] Hochreiter, S., &Schmidhuber, J. (1997). Long Short-Term Memory. Neural Computation, 9, 1735-1780.

[18]Chung, J., Gulcehre, C., Cho, K., &Bengio, Y. (2014). Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555*.

[19] Janocha, K., &Czarnecki, W. (2017). On Loss Functions for Deep Neural Networks in Classification. ArXiv, abs/1702.05659.

[20] Nasr, G.E., Badr, E., &Joun, C. (2002). Cross Entropy Error Function in Neural Networks: Forecasting Gasoline Demand. FLAIRS Conference.

[21] Baldi, P., Brunak, S., Chauvin, Y., Andersen, C.A., & Nielsen, H. (2000). Assessing the accuracy of prediction algorithms for classification: an overview. Bioinformatics, 16 5, 412-24.

[22] Vieira, S., Kaymak, U., & Sousa, J. (2010). Cohen's kappa coefficient as a performance measure for feature selection. International Conference on Fuzzy Systems, 1-8.

[23] Flach, P.A., Hernández-Orallo, J., &Ferri, C. (2011). A Coherent Interpretation of AUC as a Measure of Aggregated Classification Performance. ICML.

[24] Divyansh Agrawal (2020,April). January Flight Delay Prediction,US Flight Data for the month of Jan 2019 and Jan 2020., Version 2. Retrieved on Aug 14,2020 from https://www.kaggle.com/divyansh22/flight-delay-prediction

[25] Mia, M.M., Biswas, S.K., Urmi, M.C., & Siddique, A. (2015). An Algorithm For Training Multilayer Perceptron MLP For Image Reconstruction Using Neural Network Without Overfitting. International Journal of Scientific & Technology Research, 4, 271-275.