Airline Delay Prediction using Machine Learning and Deep Learning Techniques

Devansh Shah, Ayushi Lodaria, Danish Jain, Lynette D'Mello

Abstract*: *In this paper, we have tried to predict flight delays using different machine learning and deep learning techniques. By using such a model it can be easier to predict whether the flight will be delayed or not. Factors like 'WeatherDelay', 'NASDelay', 'Destination', 'Origin' play a vital role in this model. Using machine learning algorithms like Random Forest, Support Vector Machine (SVM) and K-Nearest Neighbors (KNN), the f1-score, precision, recall, support and accuracy have been predicted. To add to the model, Long Short-Term Memory (LSTM) RNN architecture has also been employed. In the paper, the dataset from Bureau of Transportation Statistics (BTS) of the 'Pittsburgh' is being used. The results computed from the above mentioned algorithms have been compared. Further, the results were visualized for various airlines to find maximum delay and AUC-ROC curve has been plotted for Random Forest Algorithm. The aim of our research work is to predict the delay so as to minimize loses and increase customer satisfaction.

Keywords*: *Support Vector Machine (SVM), K-Nearest Neighbors (KNN), Random Forest, Long Short-Term Memory (LSTM), RNN.

I. INTRODUCTION

The aviation industry around the globe incur huge losses due to various factors, one of these factors is Airline Delay. Airline delay tends to be onerous for every entity involved i.e. airports, airlines and passengers [1]. Precise and meticulous prediction of Airline delay using the factors which play prodigious role will be the key to minimize the losses and increase customer satisfaction. In the paper, several machine learning and deep learning algorithms have been employed to produce a comparative study with respect to the accuracy of each algorithm. As per the numbers from Bureau of Transportation Statistics (BTS) approximately 20% of all the commercial flights are delayed. BTS classifies the flights to be delayed only if they arrive 15 or more minutes late than the scheduled arrival time [2]. In this paper, every flight which is 10 or more minutes late is categorized to be 'Delayed'. BTS usually places the cancelled and diverted flights under the delayed category but to improve the accuracy and provide better results those entries have not been included while predicting flight delay. Using the 'Arr_Delay' column values an additional column 'Delayed' has been procured. The 'Delayed' column comprises of only two values – 0 and 1. These values represent the status of the flight. Flights arrived on time and after 10 minutes of scheduled arrival are represented by 0 and 1 respectively. We have then calculated precision, recall, f1-score and support using the 'Delayed' column.
Accuracy has also been calculated for Random Forest, Support Vector Machine (SVM), K-Nearest Neighbors (KNN) and Long Short-Term Memory (LSTM) which were 92.013%, 91.398%, 78.633% and 91.103% respectively. Random Forest provided the best output with maximum accuracy and thus highest reliability.
The paper is structured as follows: In section II, we have done literature review from several other papers, in section III, we have described the data preprocessing and cleaning techniques involved. The methodologies used and comparative study has been done in section IV.

II. LITERATURE REVIEW

- Choi et al. [3] proposed a prediction model to accurately predict individual flight delays. They have employed Long Short-Term Memory RNN architecture trying to prove that the accuracy increases with deeper architectures. To train the model, stochastic gradient descent (SGD) algorithm is utilized. Use of SGD helped prevent overfitting and increase general performance. The comparison of accuracies obtained with different number of layers has been formulated to support the claim of accuracy increasing with the increase in number of layers. The accuracy further improved with increasing epochs. The model has then been used to calculate and compare the delay of individual flights which manifests promising results.
- Navoneel Chakrabarty et al. [4] applied Gradient Boosting Classifier to analyze and predict possible arrival delay. Data balancing is done using Randomized SMOTE technique which in turn helped improve validation accuracy. A 200% Randomized-SMOTE is done on the dataset to reduce the imbalance between classes. There have been two strategies followed and compared throughout the paper. In strategy 1, the data imbalance removal step has been skipped and in strategy 2 it has been followed. Mean score on both the strategies were calculated and grid search graphs were depicted using the same. ROC curves and confusion matrix have also been formulated. The comparison makes it evident that data imbalance removal step results in better results.
- Roshni Musaddi et al [5] proposed a model to predict flight delays implementing Binary Classification. Their aim was to compare different flights and their delays to enable passengers to choose the apt airline before travelling. The dataset is converted into sparse matrices using label Binarizer and then the random forest algorithm is applied on the training dataset.

Revised Manuscript Received on July 21, 2020.
Devansh Shah, Department of Computer Engineering, Dwarkadas J. Sanghvi College of Engineering, University of Mumbai, Mumbai, India. Email: shah.devansh77@gmail.com
Ayushi Lodaria, Department of Computer Engineering, Dwarkadas J. Sanghvi College of Engineering, University of Mumbai, Mumbai, India. Email: ayushilodaria@gmail.com
Danish Jain, Department of Computer Engineering, Dwarkadas J. Sanghvi College of Engineering, University of Mumbai, Mumbai, India. Email: danishjain2608@gmail.com
Lynette D'Mello, Asst. Prof., Department of Computer Engineering, Dwarkadas J. Sanghvi College of Engineering, University of Mumbai, Mumbai, India. Email: lynette.dmello@djsce.ac.in

Different histograms were plotted on basis of the obtained results and the ROC Curve is plotted to show the accuracy of the model.

- Balasubramanian Thiagarajan et al. [6] proposed a two stage model to optimize prediction of flight delays. The first stage predicts the occurrence of flight delays using binary classification where Gradient Boosting Classifier gave promising results. To improve the base results feature scaling, hyper-parameter tuning and selective training are applied. Random Forest gives the most optimum results for delay and arrival prediction. The ROC Curve plotted for arrival and delay prediction gives the maximum area under the curve for Random Forest algorithm.
- Swaminathan Meenakshisundaram et al. [7] applied Logistic Regression and Decision Tree (Random Forest) algorithms on the model to predict delays. Factor analysis is used to understand the possible factors affecting the delay of a flight. Hence, the analyzed factors are implemented using the random forest algorithm. The estimate time of arrival and delays are compared from both the models. The research claims decision tree algorithm to be more effective compared to logistic regression.
- Alice Sternberg et al. [8] has done a thorough survey of a lot of approaches used to predict and analyze flight delays. A detailed timeline denoting different models used for prediction are depicted. It was verified by the authors that Normal distribution fitted better to departure delays, while Poisson distribution fitted better to arrival delays. The increase in performance and accuracy throughout the years is also mentioned.

III. PROPOSED METHODOLOGY

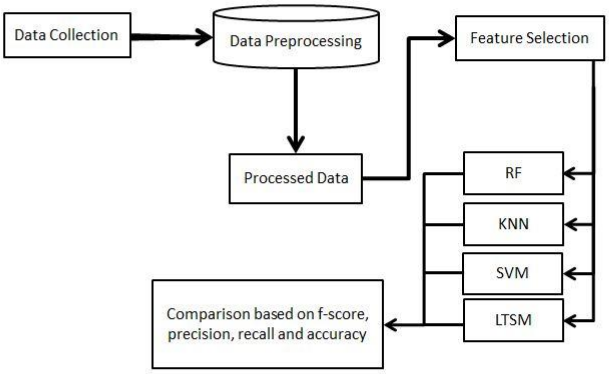

Fig 1: Proposed Methodology

This section gives a brief flowchart of the proposed framework. The first step is Data Collection, the primary and the most crucial step towards building a model. There are a few factors which are important while acquiring the dataset like the legitimacy, correctness and validity of the data. The data used here is acquired from the US Department of Transportation's Bureau of Transportation Statistics (BTS). Then comes the Data Preprocessing phase, here the data obtained is treated and changed into the desired format. Cleaning of data i.e. to remove the tuples have null values, checking for redundancies, dropping irrelevant details, etc. is carried out in this phase. To be able to work efficiently with the data, a new column 'Delay' has been added. This column categorizes all the data into two categories – on time and delayed. The output of this step is the processed data which is suitable for the algorithms to work on. The processed data is then used for extracting features which are relevant and needed for training and testing phase. The training and testing phase is carried out in each of the four algorithms used. Then Voting takes place where the values like f-score, recall, precision and accuracy of all the four algorithms are compared

IV. DATA ACQUISITION AND PREPROCESSING

A. The Dataset

- The data was collected from US Department of Transportation's Bureau of Transportation Statistics (BTS) [9]. The data to be studied was extracted from BTS having the flight information 'Year', 'Quarter', 'Month', 'AirlineID', 'UniqueCarrier', 'Origin', 'OriginCityName', 'Dest', 'DestCityName', 'Carrier', 'FlightDate', 'DayofMonth', 'DayOfWeek', 'FlightNum', 'TailNum', 'ActualElapsedTime', 'ArrDelay', 'ArrTime', 'DepDelay', 'DepTime', 'Distance', 'Cancelled', 'Diverted', 'CarrierDelay', 'WeatherDelay', 'NASDelay', 'SecurityDelay', 'LateAircraftDelay' for the year 2016.
- Due to the limitation of the machine, we had to reduce the 4,00,000 tuples and restrict our study to the airport of Pittsburgh (PIT).

B. Data Preprocessing

Before training the data, the data needs to be preprocessed so as to avoid any errors later [10]. The data has been preprocessed using various Python programming and its various libraries. The techniques employed are as follows:

1. Ignore Tuples With Null Values: In the beginning, the dataset contained 4 tuples with all values as 'NaN', so these tuples had to be dropped since they were of no use in the analysis.
2. Handling Missing Values: There were a few missing values in the 'ArrDelay' column. This could be easily replaced by the mean of the delay of that column. But there were tuples with a missing value in the 'Dest' column, so those columns had to be dropped.
3. Dropping Irrelevant Attributes*:* Most of the attributes are relevant but not all are requires so some of them were dropped.

- 'Year' has no relevance in our dataset since the entire dataset is for the same year.
- 'Quarter' and 'FlightDate' are also repetitive features so we have dropped those. Similarly, 'Origin' and 'OriginCityName' and 'Dest' and 'DestCityName' are also repetitive and have been dropped.

4. Creating Dummy Variables: This involves converting the categorical variable into dummy/indicator variable. For a categorical variable that takes on more than one value, a dummy variable is created for each unique value that the categorical variable takes on. So now all the categorical data is converted into dummy variables with values 0 and 1 (0 if not present and 1 if present).
5. Finding Delay for each Tuple: After getting a cleaned dataset, we calculated the column 'Delayed' and value 0 or 1, depending on the delay.

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

If the 'ArrDelay' is less than 10 minutes, then we assign 0, which means the flight arrived on time. If the flight is delayed more than 10 minutes, then we assign 1, which indicates that the flight is delayed.

Table 1: Classification of Flight Delay [11]

ArrDelay	Delayed
Less than 10 minutes	0
More than 10 minutes	1

V. METHODS

Random Forest: Random forest is a supervised learning algorithm which is used for both classification and regression in the field of Machine Learning. It is an ensemble learning algorithm i.e. a process of combining multiple classifiers to solve a complex problem and to improve the overall performance of the mode l [12]. Random forest are built by combining the predictions of several decision trees, each of which is trained separately.

Mathematical Representation and Equations: A tree structure is constructed, to classify the features, based on the equations and parameters. Gini Index (for any tuple S) can be used to generate the decision tree and is calculated as follows. [13]

$$Gini\ (y, S) = 1 - \Sigma_{cj\ \epsilon\ dom\ (y)} \left(\frac{|\sigma_y = cj\ S|}{|S|}\right) \quad \text{------------------ (1),}$$

Information gain and entropy can also be used to make a decision tree and get a subsequent output.

It can be calculated using the following formula given below – [14]. For any given set S, p is the proportion of S belonging to class 'i'.

$$Entropy\ (S) = \sum -p(i)log_2 p(i) \quad \text{-------------- (2)}$$

$$Gain\ (S,A) = Entropy\ (S) - \sum\frac{|Sv|}{|S|}Entropy(S_v) \quad \text{------------- (3)}$$

where, S_v = subset of S for which attribute A has value v. [15]

Support Vector Machine: Support Vector Machine (SVM) are supervised learning algorithms that analyse data used for classification and regression. SVM tries to maximise the geometric margin as well as minimize the empirical classification error [16]. It creates a set of hyperplanes in a high dimensional or infinite dimensional space, used for classification and regression. Given a set of training data set points, each marked as one or the other of two categories, an SVM algorithm builds a model that assigns new examples to one of the categories, making it a non-linear binary classifier. Parameters of maximum-margin hyperplane are derived by solving the optimization problem. A common method to solve the Quadratic Programming problem is Platt's Sequential Minimal Optimization algorithm. Quadratic Programming problem can also be solved using an interior point method that uses Newton-like iterations to find a solution of the Karush-Kuhn -Tucker conditions of the primal and dual problems [17].

K – Nearest Neighbours: K-Nearest Neighbours (kNN) is a supervised learning algorithm which can be used for pattern recognition, data mining and intrusion detection. For a data point to be classified, its k nearest neighbourhoods are retrieved to form a neighbourhood of t [q8]. A new tuple is classified using the training tuples which are similar to that tuple. There is no learning phase in kNN. When the test tuple is being classified, all needful computations are performed. In an n-dimensional space, where n dimensions are the set of n attributes which describe the dataset, the training tuple is a set of data points in this n-dimensional space. To classify an unknown tuple, the k nearest data points to it have to calculated in the n-dimensional space [19]. Various distance metrics such as Euclidean distance, Minkowski distance, and Manhattan distance can be used to find the k nearest data points to the unknown tuple. The formula for Euclidean distance between two data tuples X and Y is as given below:

$$\sqrt{\sum_{1 \le i \le n}(x_i - y_i)^2} \quad \text{.... (1)}$$

Algorithm: The pseudo code for K-Nearest Neighbours algorithm is as follows: [20]

Let m be the number of training data samples.

Let p be the unknown point.

Step 1: *Initialize an empty array (say a[]). Store all the training data samples in this array*

Step 2: *For i=0 to m:*

Using equation (1), Calculate the Euclidean Distance d(a[i],p).

Step 3: *The K smallest distances obtained are stored in set S. Each of these distances corresponds to a point which is already classified.*

Step 4: *Return the majority label amongst S.*

Long Short-Term Memory: Long Short-Term Memory (LSTM) is used in deep learning. It is an artificial recurrent neural network (RNN) architecture. An LSTM unit is devised of a cell, an input gate, an output gate and a forget gate. The flow into the cell is regulated by the input gate, the degree to which a value remains in the cell is decided by the forget gate and the degree to which the value in the cell is used for the calculation of the output activation is decided by the output gate [21]. Logistic sigmoid function is usually the chosen activation function of LSTM gates.

Architecture: LSTM's contain special units called memory blocks in the hidden recurrent layer [22]. Each memory block contains and input and output gate in the original architecture. The input gate regulates the flow of input activations into the memory cell. The output gate regulates the output flow of cell activations into the rest of the network.

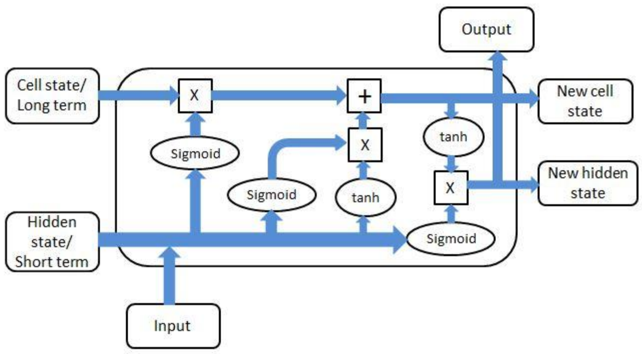

Figure 2: LSTM Architecture

VI. RESULTS

A. Data Visualization

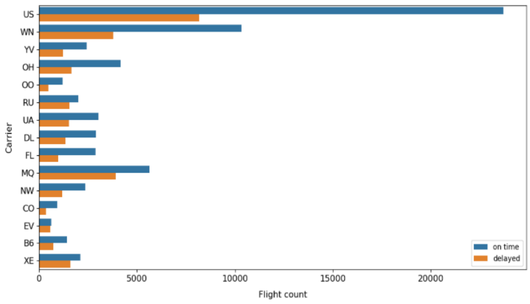

Figure 3: Carrier vs Flight count

In Figure 3, we represent the number of flights that arrive on time or are delayed for a certain carrier. The blue bar shows the number of flights arriving on time whereas the orange bar shows the number of flights which are delayed.

We can evidently observe that the number of flights on time for the carrier USAirways (US) has maximum flights flying. It has maximum number of flights that arrive on time, followed by SouthWest Airlines (WN).

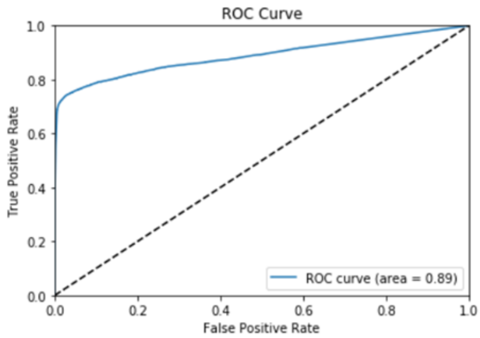

Figure 4: AUC-ROC Curve Random Forest Algorithm

An AUC-ROC Curve is a way to measure the performance for classification problem at various level of thresholds. [23] Receiver Operating Characteristic (ROC) curve shows the probability at various thresholds and the Area Under the Curve (AUC) represents how capable the model is to distinguish between various classes. Higher the AUC, better the model is at predicting the correct value (i.e. 0's as 0 and 1's as 1). Figure 4 shows The Area Under the Curve shown for Random Forest which is 0.89. It is an admissible value since it tends towards 1, so the performance of the model will be better.

B. CONFUSION MATRIX/ CROSS TAB FOR AIRLINE DELAY ASSESSMENT

The Accuracy Score, after testing the model, subsequent to the training phase, is equal to approximately 92.013%. This means that the machine learning model classified 92.013% of the testing data tuples correctly. The results of this are displayed below in the result section of this paper. We can see that the diagonal elements of the above confusion matrix is the total number of tuples correctly classified [1] [2]. The Confusion Matrix is as follows:

```
Confusion Matrix :
[[21488   632]
 [ 1643  4721]]
```

Figure 5: Confusion Matrix generated for Random Forest algorithm

$$\text{Accuracy} = \frac{\textit{Total number of Correctly Classified Tuples}}{\textit{Total Number of Tuples}}$$

$$\text{Accuracy} = \frac{21488+4721}{21488+632+1643+4721} = \frac{26209}{28484} = 0.92013$$

Accuracy Score = 92.0123% ---*(7)*

```
Random Forest:
no of training samples =  66460
no of testing samples =  28484
ACCURACY :       92.01305996348827

Report :
                 precision    recall  f1-score   support

           0       0.93      0.97      0.95     22120
           1       0.88      0.74      0.81      6364

    accuracy                           0.92     28484
   macro avg       0.91      0.86      0.88     28484
weighted avg       0.92      0.92      0.92     28484
```

Fig 6: Accuracy, precision, recall, f1-score, support of Random Forest Algorithm

C. Comparison between the algorithms used

Random Forest, SVM, kNN classifier and LSTM algortihms are applied on a dataset using Python programming language, to classify if a flight is delayed by more than 10 minutes based on characteristics such as Origin, Destination, NASDelay,WeatherDelay,LateAircraftDelay,Month,etc.

Thus we use the '*Delayed*' attribute of our dataset, as the target variable to measure and categorise aircraft delay.

1) Delay Classification: Parameters like Origin, Destination, NASDelay, WeatherDelay our used to train the model against '*Delayed*' attribute of the dataset. Thus, performing different algorithms on the dataset, we obtain – *No. of Training Samples = 66460; No. of Testing Samples = 28484.* The model performance is evaluated on the following metrics as follows [24]:

- Validation accuracy describes how correctly the model predicts the samples in the test set.
- Recall is defined as the fraction of relevant instances retrieved to the total amount of relevant instances.
  Recall = TP/(TP+FN)
- Precision is defined as the fraction of correctly predicted positive observations (relevant instances) to the total predicted positive observations (total retrieved instances)
  Precision = TP/(TP +FP)
- F1-Score is a statistical measure of accuracy and defined as harmonic mean or the weighted average of precision and recall.

Table – 2: Comparison table of results obtained

	Random Forest	SVM	kNN	LSTM
Accuracy	92.013%	91.398%	78.633%	91.864%
Recall	0.86	0.83	0.53	0.93
F1-Score	0.88	0.86	0.50	0.82
Precision	0.91	0.92	0.78	0.76

Table - 2 given above summarizes the *accuracy, precision, recall and f1-score* obtained from various algorithms which have been researched.

Thus, given any set of attributes (feature values), the **Random Forest** machine learning model can accurately predict if an aircraft travelling from a specific origin to a destination with a specified set of parameters will arrive on time or get delayed, with an accuracy of approximately 92.013%. An accuracy of 92.013% succinctly proves the efficiency of this model, for our purposes. As it can be observed from the confusion matrix, very few tuples of the testing dataset have been incorrectly classified. Thus, this satiates our requirement of determining the delay for any given aircraft, given merely the parameters of it.

VII. CONCLUSION

The above study shows that machine learning and deep learning algorithms can be efficaciously used to predict flight delays. The purpose of doing the above classification and analysis, is to gauge the delay not only to suffice the various purposes of mankind, but also analyze factors affecting delay such as 'WeatherDelay', 'NASDelay'. The best results are obtained by the Random Forest based model which provides 92.023% accuracy in classifying delay, into 2 categories based on the above-mentioned parameters like Origin, Destination, Month, etc. Thus, by using Random Forest based model the flight delay can be predicted, which will be beneficial for all the entities involved i.e. airport, airline and passengers. Therefore, the analysis of flight delay carried out through this paper is based solely on scientific parameters and is of paramount importance in the aviation industry.

Future Scope: The above research methodology should be performed on the data collected for the recent years, owing to the population rise in recent years leading to increase in the number of flights. To obtain a detailed analysis, a more thorough localized search and scrutiny must be conducted to accurately determine the arrival or departure delay. Moreover, this methodology can be used for all the airports. The results of our research can be extrapolated to perform the above and determine accurately the delay and help in determining the major reasons causing it.

REFERENCES

1. Yogita Borse , Dhruvin Jain , Shreyash Sharma , Viral Vora, Aakash Zaveri, 2020, Flight Delay Prediction System, International Journal Of Engineering Research & Technology (IJERT) Volume 09, Issue 03 (March 2020).
2. Ye, B.; Liu, B.; Tian, Y.; Wan, L. A Methodology for Predicting Aggregate Flight Departure Delays in Airports Based on Supervised Learning. *Sustainability* 2020, *12*, 2749.
3. Y. J. Kim, S. Choi, S. Briceno and D. Mavris, "A deep learning approach to flight delay prediction," 2016 IEEE/AIAA 35th Digital Avionics Systems Conference (DASC), Sacramento, CA, 2016, pp. 1-6, doi: 10.1109/DASC.2016.7778092.
4. Chakrabarty, Navoneel. "A Data Mining Approach to Flight Arrival Delay Prediction for American Airlines." *2019 9th Annual Information Technology, Electromechanical Engineering and Microelectronics Conference (IEMECON)* (2019): 102-107.
5. Musaddi, Roshni & Jaiswal, Anny & Girdonia, Mansvi & Sanjudharan, M S Minu. (2018). Flight Delay Prediction using Binary Classification. 6. 34-38.
6. B. Thiagarajan, L. Srinivasan, A. V. Sharma, D. Sreekanthan and V. Vijayaraghavan, "A machine learning approach for prediction of on-time performance of flights," 2017 IEEE/AIAA 36th Digital Avionics Systems Conference (DASC), St. Petersburg, FL, 2017, pp. 1-6, doi: 10.1109/DASC.2017.8102138.
7. V. Natarajan, S. Meenakshisundaram, G. Balasubramanian and S. Sinha, "A Novel Approach: Airline Delay Prediction Using Machine Learning," 2018 International Conference on Computational Science and Computational Intelligence (CSCI), Las Vegas, NV, USA, 2018, pp. 1081-1086, doi: 10.1109/CSCI46756.2018.00210.
8. Sternberg, Alice & Soares, Jorge & Carvalho, Diego & Ogasawara, Eduardo. (2017). A Review on Flight Delay Prediction.
9. Bureau of Transportation Statistics. Available online: https://www.bts.gov/
10. Chakrabarty, Navoneel. "A Data Mining Approach to Flight Arrival Delay Prediction for American Airlines." *2019 9th Annual Information Technology, Electromechanical Engineering and Microelectronics Conference (IEMECON)* (2019): 102-107.
11. V. Venkatesh, A. Arya, P. Agarwal, S. Lakshmi and S. Balana, "Iterative machine and deep learning approach for aviation delay prediction," 2017 4th IEEE Uttar Pradesh Section International Conference on Electrical, Computer and Electronics (UPCON), Mathura, 2017, pp. 562-567, doi: 10.1109/UPCON.2017.8251111.
12. Leo Breiman - Random Forests (dept. Of statistics, University of California, Berkeley)
13. Rokach, Lior & Maimon, Oded. (2005) - Decision Trees. 10.1007/0-387-25465-X_9.
14. Yang, Ning & Li, Tianrui & Song, Jing. (2007). Construction of Decision Trees based Entropy and Rough Sets under Tolerance Relation. International Journal of Computational Intelligence Systems. 10.2991/iske.2007.258.
15. Liu Y., Wang Y., Zhang J. (2012) New Machine Learning Algorithm: Random Forest. In: Liu B., Ma M., Chang J. (eds) Information Computing and Applications. ICICA 2012. Lecture Notes in Computer Science, vol 7473. Springer, Berlin, Heidelberg
16. Durgesh K.Srivastava, Lekha Bhambu – Data Classification using Support Vector Machine
17. B. E. Boser, I. M. Guyon, and V. N. Vapnik. A training algorithm for optimal margin classifiers. In D. Haussler, editor, 5th Annual ACM Workshop on COLT, pages 144-152, Pittsburgh, PA, 1992. ACM Press
18. Guo, Gongde & Wang, Hui & Bell, David & Bi, Yaxin. (2004). KNN Model-Based Approach in Classification.
19. Alka Lamba, Dharmender Kumar. Survey on KNN and its Variants. International Journal of Advanced Research in Computer and Communication Engineering Vol.5,Issue 5, May 2016.
20. Geeks For Geeks – Available: https://www.geeksforgeeks.org/k-nearest-neighbours/
21. Sherstinsky, Alex. "Fundamentals of Recurrent Neural Network (RNN) and Long Short-Term Memory (LSTM) Network." Physica D: Nonlinear Phenomena 404 (2020): 132306. Crossref. Web.
22. Hasim Sak, Andrew Senior, Francoise Beaufays. Long Short-Term Memory Recurrent Neural Network Architectures for Large Scale Acoustic Modeling
23. Towards DataScience – Available: https://towardsdatascience.com/understanding-auc-roc-curve-68b2303cc9c5
24. Available - https://blog.exsilio.com/all/accuracy-precision-recall-f1-score-interpretation-of-performance-measures/

AUTHORS PROFILE

Devansh Shah is an undergraduate student pursuing computer engineering at Dwarkadas J. Sanghvi College of Engineering, Mumbai University, India. Work experience includes working at start-up as a Data Science and Artificial Intelligence intern.

Ayushi Lodaria is an undergraduate student pursuing computer engineering at Dwarkadas J. Sanghvi College of Engineering, Mumbai University, India. Her interests include Machine Learning and Data Analytics.

Danish Jain is an undergraduate student pursuing computer engineering at Dwarkadas J. Sanghvi College of Engineering, Mumbai University, India. His interests include Machine Learning, Deep Learning and Artificial Intelligence.

Mrs. Lynette Dmello, Asst. Professor in Department of Computer Engineering at D J Sanghvi College of Engineering, Mumbai. Having M.E degree in Computer Engineering from Mumbai University. Her research interest includes Natural language processing, Data Analytics, Compiler optimization.