FIDDLE TOUR: EFFICIENT TRAJECTORY BASED FRAUDULENT TAXI TRIP DETECTION USING ADABOOST AND XGBOOST

A PROJECT REPORT

Submitted by

RAMYA.S.P (1815024)

USHEKHA.U (1815042)

in partial fulfilment for the award of the degree

of

BACHELOR OF TECHNOLOGY

in

INFORMATION TECHNOLOGY

NATIONAL ENGINEERING COLLEGE

K.R.NAGAR, KOVILPATTI- 628 503

ANNA UNIVERSITY:: CHENNAI 600 025

APRIL 2022

ANNA UNIVERSITY: CHENNAI 600 025

BONAFIDE CERTIFICATE

Certified that this project report **"FIDDLE TOUR: EFFICIENT TRAJECTORY-BASED FRAUDULENT TAXI TRIP DETECTION USING ADABOOST AND XGBOOST"** is the bonafide work of **S.P.RAMYA (1815024)** and **U.USHEKHA (1815042)**, who carried out the project under my supervision.

SIGNATURE
Dr.K.G.SRINIVASAGAN
HEAD OF THE DEPARTMENT
Department of Information Technology
National Engineering College
K.R.Nagar, Kovilpatti – 628 503.

SIGNATURE
Dr.R.MUTHUKKUMAR
SUPERVISOR
Associate Professor
Department of Information Technology
National Engineering College
K.R.Nagar, Kovilpatti – 628 503.

Submitted to the viva-voce examination held at **NATIONAL ENGINEERING COLLEGE, K.R.NAGAR, KOVILPATTI – 628 503** on _____.

INTERNAL EXAMINER **EXTERNAL EXAMINER**

ABSTRACT

Taxi service is playing a vital role in the public transportation system across the world. In India, most of the cities use an important taxi service as transport system and it is often corrupted by fraud and passengers are paying high fares while traveling in a taxi. A passenger being overcharged by the taxi driver is a type of fraudulent trip, and it brings negative impacts to smart city application. The existing system relies on the assumption that the trip is correctly determined by the taximeter. Many taxi drivers are indulged in fraudulent activities by carrying passengers without activating the taximeter and trying to get more fare from the passengers. Taking the longest route possible by the taxi drivers is another scam tactic to make a few extra rupees. To overcome these above challenges fiddle tour framework is proposed for detecting fraudulent taxi trips while traveling in a taxi. The proposed approach uses machine learning algorithms like AdaBoost and XGBoost for increasing fraudulent taxi trip detection very accurately. The fraud trip is recognised when the passenger is requested for various charges in the total amount. It also predicts the unmetered taxi trips based on trajectories and detects overcharging while traveling. The experimental results reveal that the proposed approach outperforms other existing approaches in terms of efficient and accurate fraud detection.

ACKNOWLEDGEMENT

First of all, we express our heartfelt thanks to the almighty for his blessings. We find immense pleasure to convey our sincere and grateful thanks to our Management and respected Director **Dr.S.Shanmugavel, B.Sc., D.M.I.T., Ph.D.,** for being our inspiration in undertaking and fulfilling the task.

We thank our beloved Principal **Dr.K.Kalidasa Murugavel, M.E., Ph.D., FIE.,** for providing the necessary facilities in carrying out this project work on our campus.

We sincerely thank **Dr.K.G.Srinivasagan, M.E., Ph.D., Professor and Head**, Department of Information Technology, for his moral support in bringing up this project.

We are dutiful to our guide **Dr.R.Muthukkumar, M.E., Ph.D., Associate Professor**, Department of Information Technology, for his valuable guidance. We also thank him for his constant encouragement and guidance in bringing out this project a success manner.

We express our regards and sincere thanks to our project coordinator and **Mrs.N.Gowthami, M.E., Assistant Professor**, Department of Information Technology for their continuous effort to complete this work successfully.

We extend our thanks to all the faculty members and non-teaching staff members for their timely help and support. We are grateful to thank all our friends for their valuable feedback and suggestions. Finally, we thank our parents for their support in making this project a success.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF ABBREVIATIONS

C-IOV	Collaborative-Internet of Vehicles
CPD	Collaborative Preference Discover
DIS	Difference and intersection set
GBM	Gradient Boosting Machines
GPS	Global Positioning System
IERP	Improved Edit Distance with Real Penalty
IOV	Internet of Vehicles
MDP	Markov Decision Process
ML	Machine Learning
NYC	New York City
OD	Origin Destination
TL	Transfer Learning
V2V	Vehicle to Vehicles

CHAPTER 1

INTRODUCTION

This chapter gives a general introduction to Taxi services and driver behaviors in modern cities, benefits, and convenience to our daily life. A brief review of literature about the present work is also presented.

1.1. TAXIS' DRIVING BEHAVIORS

Taxi is major transportation in the urban area, offering great benefits and convenience to our daily life. However, one of the major business frauds in taxis is the charging fraud, specifically overcharging for the actual distance. In practice, it is hard for us to always monitor taxis and detect such fraud. Due to the global positioning system (GPS) embedded in taxis, we can collect the GPS reports from the taxis' locations, and thus, we can retrieve their traces. Intuitively, we can utilize such information to construct taxis' trajectories, compute the actual service distance on the city map, and detect fraudulent behaviors. However, in practice, due to the extremely limited reports, notable location errors, complex city maps, and road networks, our task to detect taxi fraud face significant challenges, and the previous methods cannot work well.

Taxi services in modern cities are often corrupted by fraud, and passengers are often overcharged by taxi drivers. It is difficult to find public transportation services at the midnight and prohibited to drive after drinking. These taxi frauds result in many complaints and may lead to a bad reputation for taxi services.

Common taxi frauds include: taximeter tampering, detour, and refusing of service. These fraudulent behaviors usually have evident properties that differ from normal taxi trips. For example, the distance of a detour is usually longer than normal paths

between a pair of source and destination; the reported speed of the taxi with a tampered taximeter tends to be higher than its actual speed, and the taxi drivers with a higher income are more likely to refuse passengers traveling to unpopular areas. Many existing methods try to detect these types of behaviors.

As IoT moves toward smart transportation, smart industry, smart health, and other various industries, which makes cities smarter. The Internet of Vehicles (IoV), an important branch of the IoT in the field of smart transportation, is a new industrial form of the transportation industry that deeply integrates electronics, information communication, transportation, artificial intelligence, etc., and is an effective method for urban vehicle trajectory supervision. However, IoV has problems of inconsistency and difficulty in integrating multicategory information, especially the large number of trajectories produced by taxis. Accurately detecting, identifying, and correcting taxi drivers' outlier trajectories are of great significance to improving government traffic management, industry service quality, and passenger travel experience. Thence, the Collaborative Internet of Vehicles (C-IoVs) is considered to be the solution to these problems. C-IoVs can connect roadside units (RSU) deployed on traffic roads to achieve information exchange through wireless communication technology, including vehicle-to-vehicle (V2V), vehicle-to-infrastructure, etc., which can facilitate the modeling, monitoring, and optimizing of the whole process in outlier driver trajectories accurately.

Recent years have witnessed the ever-increasing urban vehicles, which has brought many problems such as traffic congestion, energy consumption, and environmental pollution. Private cars, a class of small motor vehicles usually registered by individuals and for personal use, constitute a large proportion of urban vehicles. the regular travel behaviors of private cars have considerable impacts on urban traffic, which contributes significantly to congestion and the formation of urban hot zones. In particular, there are notable differences in the travel needs and travel

behaviors when people drive private cars compared with floating cars, i.e., buses, taxis, and ride-hailing vehicles.

1.2. TAXI FRAUD BEHAVIORS

Taxi fraud is not a special phenomenon that is only existing in many countries but one that is always happening in urban transportation. It is also a severe problem for the city traffic management. Usually, the more complex the traffic situation of a city, the more taxi fraud. Thus, the threat of taxi fraud is particularly rampant in those cities with huge populations and complicated road networks. Authorities nowadays have paid a lot of attention and effort to this practical and serious problem.

For example, to prohibit taxi fraud, the local authorities adopt extensive random-selective examinations periodically. However, this scheme takes tremendous human resources and economic costs. Moreover, the selective examination is not efficient and effective. Fortunately, the GPS device is equipped in the taxi, reporting taxis' traces to the companies or authorities. Based on the taxis' traces and the digital road map, hopefully, we can retrieve the reported trajectory and compare it with the actual trajectory to detect the overcharged distances of the taxis. Such distance-based methods are based on map matching and distance-based clustering.

However, there exist notable location errors and extremely limited reports in the GPS localization data. Such practical issues make it hard for us to retrieve accurate trajectories and then effectively conduct taxi fraud detection. At the same time, the complex road networks and dynamic information in the geographical data make the previous methods face big challenges in scalability, data storage, and updating. Based on careful investigation of taxi fraud behaviors, we noticed that fraud taxi drivers always modify the taximeter to a smaller scale, which means that the taximeter records a longer distance than the actual distance.

As a result, the reported speed of the taxi is higher than the actual speed. At the same time, fortunately, the reported speed from the GPS record on a taxi is accurate because it is directly collected from each vehicle's speedometer, not from the GPS measurement. Hence, taking advantage of the given interesting observations on the relationship between the speed information and the fraud behavior, we present a novel method to model taxis' driving behaviors and learn this model from a large-scale real-life data set.

1.3. TAXI SHARING

A taxi-sharing system that accepts taxi passengers' real-time ride requests sent from smartphones and schedules proper taxis to pick up them via taxi-sharing with time, capacity, and monetary constraints (the monetary constraints guarantee that passengers pay less and drivers earn more compared with no taxi-sharing is used). Our system saves energy consumption and eases traffic congestion while enhancing the capacity of commuting by taxis. Meanwhile, it reduces the taxi fare of taxi riders and increases the profit of taxi drivers. real-time taxi-sharing has not been well explored, though ridesharing based on private cars, often known as carpooling or recurring ridesharing, was studied for years to deal with people's routine commutes, e.g., from home to work. In contrast to existing ridesharing, real-time taxi-sharing is more challenging because both ride requests and positions of taxis are highly dynamic and difficult to predict. Passengers are often lazy to plan a taxi trip and usually submit a ride request shortly before the departure. A taxi constantly travels on roads, picking up and dropping off passengers. Its destination depends on that of passengers, while passengers could go anywhere in a city. We place our problem in a practical setting by exploiting a real city road network and the enormous historical taxi trajectory data. Compared to existing carpooling systems, the proposed ridesharing model considers more practical. constraints which include time windows, capacity, and monetary constraints for taxi trips. In addition, our work proposes efficient searching and scheduling

algorithms that are capable of allocating the “right” taxi among tens of thousands of taxis for a query in milliseconds.

1.4. MACHINE LEARNING ALGORITHM

Machine learning has been used to improve the performance of applications in many areas, including robotics, natural language processing, cyber-physical systems, networking, and intelligent transportation systems. Regression is a widely used supervised machine learning technique, which is used to predict a continuous dependent variable from several independent variables.

1.5. TAXI FRAUD DETECTION

For taxi fraud detection, A scenario of taxi fraud is where taxi drivers deliberately take unnecessary detours to overcharge passengers. They transformed the taxi fraud detection problem into finding anomalous trajectories from all the trajectories with the same source-destination pairs and used the spatial distance as the main feature to design the anomaly detection method. Another case is where fraud taxi drivers modify the taximeters to a smaller scale so that they record longer distances than the actual ones. They modeled taxi fraud behaviors in a trajectory-free and map-free scenario, constructed a model by the speed information instead of location or distance, and proposed a speed-based clustering method to detect taxi fraud. Different from the above taxi fraud scenarios, we target the unmetered taxi trips. To detect the fraud trip for unmetered trajectory the existing system uses the map matching algorithm and many unmetered taxi trips are still unidentified. The outliers are detected with the help of GPS and the taxi trip which involves complex routes decreases the accuracy. In detecting the anomalous trajectories for large taxi data sets, many sample data are left out or not considered. In the real-time route recommendation of electric taxi driving the waiting time and net revenue for the taxi are not taken into account. In those previous problems, the price for the trip is not predicted and the network routes are

not considered. To overcome these problems and to provide the efficiency of the system.

The contributions of this project are as follows.

- We detect fraudulent taxi trips using the AdaBoost and XGBoost algorithm.
- We enhance the accuracy of the trip with the help of trajectory and detect whether extra tariffs are charged or not.
- We have determined the better performance of the system by comparing AdaBoost and XGBoost using the performance metrics like precision score, recall score, F1 score, cohen's kappa, and the execution time with the efficient classifier we predict the fraud trip.

The remainder of this project report is organized as follows. We first perform the Literature Survey in Section II and then describe the proposed methodology for taxi fraud trip detection in Section III. Finally, we evaluate the system in Section IV and conclude this report in Section V.

CHAPTER 2

LITERATURE SURVEY

2.1. INTRODUCTION

This chapter reviews the various studies carried out in the design of taxi fraudulent trip detection systems. The existing taxi fraud detection system has been applied to find anomalous trajectories. The review also focuses in detail on taxi fraud detection systems for better effectiveness and efficiency of Fraud Trip on both the synthetic and real-world taxi trajectory data sets.

2.2. TAXI FRAUDULENT TRIP DETECTION SYSTEM

Ding *et al*. [1]'s system detects "unmetered" taxi trips based on a novel fraud detection algorithm and a heuristic maximum fraudulent trajectory construction algorithm. In this system, taxi drivers could "negotiate" and often overcharge passengers with a higher fare than usual. Unmetered taxi trip is a serious problem in modern cities. FraudTrip can effectively and efficiently detect fraudulent trips without the help of taximeters. Kong *et al.* [2]'s method has detected taxi driving fraud and considered the geographic constraints of the taxi trajectory and the collaboration between taxi trajectories in Collaborative Internet of Vehicles (C-IoVs). A method that combines the identified normal trajectory with environmental perception under C-IoVs to classify trajectories is affected by objective geographic factors. A conflict evidence fusion algorithm recognizes the geographic constraints of the taxi trajectory and the collaboration between taxi trajectories in C-IoVs. The geographic boundaries of the taxi trajectory and the collaboration between taxi trajectories are calculated. The system uses the taxi trajectories of C-IoVs as a benchmark to generate a pair of typical origin-destination node trajectories. The outlier trajectory is then created by combining

the three aspects of time, distance, and trip cost. The deep analysis is done for the anomalous trajectory by combining various aspects of regional anomalies.

Xiao *et al.* [3] using trajectory data analysis, concentrated on extracting the usual travel behavior of private cars. To record the temporal-spatial distance between the trajectories, the Improved Edit distance with Real Penalty (IERP) technique was used. The utilization of the stay time to quantify similarity in the time domain of the trajectories performs well. To build the trajectory similarity matrix for each private car on top of IERP is used which allows for analysis of its regular travel pattern. The Kernel Principal Component Analysis is applied to lower the data dimension of the trajectory similarity matrix and to sample the data. A transfer learning (TL)-based classification system TrAdaBoost is used to define the regular travel behavior of private cars. The challenge of artificial recognition of private cars with predictable trip patterns is transformed into model prediction using the TL approach, which considerably decreases the artificial cost and solves the problem of insufficient labeled data. The labeled and unlabeled data are combined as training data in this way.

They focused on unlabeled data and look for a suitable case in the labeled collection. Comparative experiments were conducted with existing methods for determining the trajectory distance. The findings suggest that the IERP is effective at determining the distance between spatiotemporal trajectories. Furthermore, experiments using real-world large-scale private car trajectory data show that the method outperforms well-performing learning methods such as Support Vector Machines (SVM), Multilayer Perception Classifier (MLPC), Label Propagation (LP), and transductive SVM (TSVM) in identifying the travel regularity of private cars. The use case highlights the value of extracting trajectories' regular behavior and analyzed the novel way of looking at human travel behavior when people drive their cars. The insights on normal travel behavior help people to have a better travel experience. Further, by recognizing regular travel behaviors, the

geographical distribution characteristics of people's travel can be correctly forecasted which is significant for urban planning. Liu *et al.* [4] proposed the fact that fraudulent taxis always use a secret method, such as altering the taximeter to a smaller scale, which is a critical and noteworthy finding. As a result, not only does the service distance increase, but the claimed taxi speed also increases. The first to predict taxi fraud behaviors in a trajectory-free and map-free environment, using speed data rather than position or distance to build a model. We create a probability model to detect taxi fraud using a unique speed-based clustering method to characterize taxi driving behavior. Synthetic and large-scale real data are used to create a well-designed system. These methods outperform the state of the art in terms of scalability, efficiency, and efficacy, according to the findings of the empirical experiments. The models are now being used in a city as a taxi fraud system. Not just synthetic data, but also real data, were used to assess the project. In terms of scalability, accuracy, and efficiency, the methodologies outperform the current state of the art.

Ma *et al.* [5] conceived and constructed a taxi-sharing system that receives real¬time ride requests from taxi clients via smartphone and arranges appropriate taxis to pick them up via ridesharing, subject to time, capacity, and monetary limits. The monetary constraints provide incentives for both passengers and taxi drivers: passengers will not pay more than they would if they did not use ridesharing and will be compensated if their travel time is extended as a result of ridesharing; taxi drivers will be compensated for all detours distance incurred as a result of ridesharing. While such a system is beneficial to both society and the environment.

Report on a mobile cloud-based system that enables real-time taxi-sharing in a practical context. Using an App loaded on their smartphones, cab drivers decide when to join and leave the service on their own. Extensive tests were conducted to confirm the usefulness of taxi-sharing as well as the proposed system's efficiency

and scalability. According to the results of the trial, the percentage of ride requests that are fulfilled increases by three times, while users save 7% on taxi fares by using taxi sharing when taxis are in high demand. Further, if taxi-sharing is permitted in Beijing, 2 million liters of gasoline can be saved each year. The findings also imply that, in practice, reordering the points of a schedule before inserting a new trip request is not necessary for minimizing journey distance.

Kamal *et al.* [6] introduce a method for understanding traffic circumstances that can be utilized to implement highly anticipative driving. Anticipatory driving is defined as the predictive control of a host vehicle in an expanded view, taking into account previous traffic conditions. A road-speed profile is proposed that concisely describes the mean speed in each small segment or cell of the road by effectively extracting information from traffic big data, i.e., broadcasted data from all surrounding vehicles to the host vehicle, for improved perception of traffic conditions on the road. An anticipative driving system of a host vehicle is evaluated based on the estimated speed profile.

He *et al.* [7] present a collaborative route discovery technique that takes advantage of taxi drivers' experience and preferences in urban locations. Collaborative preference discovery (CPD) and intelligent driving network development are the two primary phases (IDNG). CPD is proposed in the first phase, which involves cluster-to-cluster retrieval to capture the top-k routes that are not only frequently traversed by taxis but also neighboring to the origin-destination (OD) pair, given an OD pair and provided that the cluster is a road segment set within a time-reachable range. They can accomplish acceptable route recommendations by just evaluating a refined partial graph with highly collaborative routes rather than a global graph since experiential routes are well-organized.

Kong *et al.* [8] offered a step-by-step approach for creating a social vehicular mobility dataset using floating automobile data, which has the benefit of universality. They anticipate the OD matrix of social vehicles with the gravity

model, and then calibrate the OD matrix with the average growth factor approach, using deep analysis and modeling of the dataset of floating cars and integrating it with official data. Making it feasible to better manage public transportation. For example, using road detecting technology, we can continuously monitor traffic conditions, which can assist us in planning public transportation routes. Making it possible to follow the status of commodities in real-time. Vehicles select a path using a simulation program, which may cause traffic congestion on particular highways.

Rossi *et al.* [9] developed a model using taxi drivers' behavior and geographic data for an intriguing and difficult task: predicting the next destination in a taxi voyage. Predicting the next location is a well-studied subject in human mobility that has a variety of real-world applications, ranging from improving the efficiency of electronic dispatching systems to predicting and minimizing traffic congestion. Recurrent Neural Network technique that uses geographical information from Location-Based Social Networks to model taxi drivers' behavior and encode the semantics of visited locations. The method is unable to take advantage of travel utility aspects such as distance traveled. Because they are not available at the time of the projection, cost and journey time are not included.

Tseng *et al.* [10] In comparison to traditional taxis with internal combustion engines, the viability of electric taxis was demonstrated with the use of taxi service strategy optimization. A massive data set of real-world cab trips in New York City (NYC) is used in a big data study. This solution entails first modeling the computerized taxi service strategy using a Markov decision process (MDP) and then optimizing the taxi service strategy using data from NYC taxi trips. Electric taxi drivers' profitability is investigated empirically under various battery capacities and charging situations. an MDP to model computerized electric taxi service strategies, with explicit consideration of EV constraints such as battery capacity and charging station locations, to obtain the MDP's optimal policy based

on a big data study using a large dataset of real-world taxi trips in NYC, the impact of factors such as battery capacity and charging modes, and charging station locations on the net revenues of electric taxi drivers

Wang *et al.* [11]'s method has detected and classified the anomalous trajectory. An anomalous trajectory differs locally or globally from most other normal trajectories. The goal of anomalous trajectory identification is to locate trajectories that are different from the typical trajectories in a dataset, as well as to determine how different they are. The difference and intersection set (DIS) distance is a novel distance metric that may be used to compare any two trajectories. The DIS distance is a useful tool for determining the difference between abnormal and normal trajectories.

CHAPTER 3

EFFICIENT FRAUDULENT TAXI TRIP DETECTION USING ADABOOST AND XGBOOST

3.1. INTRODUCTION

In this section, we first formalize the methodology of our proposed system with the process of using algorithm and flow of proposed work.

3.2. EFFICIENT FRAUDULENT TAXI TRIP DETECTION

In the proposed work, the project is carried out in two phases they are the training and the testing phases. In the training Phase, the taxi trip dataset is imported and particular features are selected from the dataset to perform the classification process. After the feature is selected from the dataset, AdaBoost and XGBoost algorithm is implemented. By the classification of each classifier, the Confusion matrix is drawn to analyze the efficiency of each classifier. With the help of the confusion matrix and the performance metrics are the precision score Recall score, F1 score, and cohen's kappa the accuracy is calculated. ROC curve and Performance bar graph are drawn for both AdaBoost and XGBoost classifiers to show their accuracy of them. By the end of this phase, the best classifier is known with the help of this classifier the testing phase has proceeded. In the testing phase to find the taxi fraud trip analysis is done by using the features like pick_up location and drop_off location, payment method, and the amount asked by the taxi driver with the help of a classifier the efficiency of finding the taxi fraud trip is known.

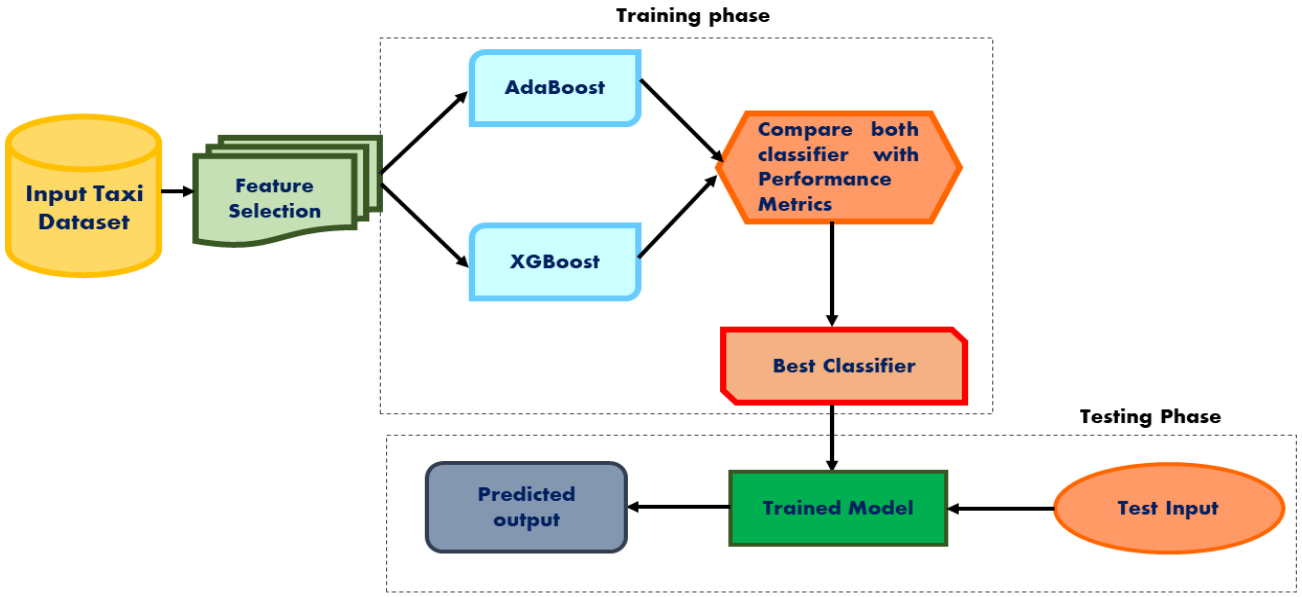


Figure 3.1 Efficient Fraudulent Taxi Trip Detection Architecture

The above flow diagram depicts the processes and steps involved in developing a fraud detection model.

3.3. ADABOOST ALGORITHM

Boosting is a broad learning approach that involves merging simpler classifiers. The concept of boosting is the process of taking a "weak classifier" one that does at least marginally better than chance and using it to create a much stronger classifier, hence improving the performance of the system.The categorization algorithm is ineffective. This is accomplished by averaging the outputs of a number of different algorithms.The most widely used boosting algorithm is AdaBoost, which gets its name from the word "adaptive." provides very effective outcomes There's also a lot of leeway when it comes to selecting a weak classifier. Boosting is a subset of ensemble methods, a broad category of learning algorithms that aims to improve learning algorithms by mixing several smaller algorithms.

Adaboost

Given:$(x1;y1),\ldots\ldots,(x_m;y_m),x_i\in X,y_i\in Y=\{0,1\}$.

Initialize $D_1(i)=1/m$.

For t=1…T:

1.Train weak classifier using distribution D_t

2.Get weak hypothesis h_t:X→{0,1} with error $\varepsilon t=\Sigma_{i:h_t(x_i)\neq y_i}D_t(x_i)$

3. Choose $\alpha_t=\frac{1}{2}log\left(\frac{1-\varepsilon_t}{\varepsilon_t}\right)$

4.Update:

$$D_{t+1}(i)=\frac{D_t(i)}{z_t}=\begin{cases} e^{-\alpha_t}\ if\ instance\ i\ is\ correctly\ classified \\ e^{\alpha_t}\ if\ instance\ i\ is\ not\ correctly\ classified \end{cases}$$

Where Z_t is a normalization factor (chosen so that $\Sigma_{i=1}^{m}$ D_{t+1}=1)

Output the final hypothesis:H(x)= $sign(\sum_{t=1}^{T}\alpha_t h_t(x))$

a weak classifier is considered by the decision stumps in case of decision trees. Because the number of possible parameter settings is relatively small, a decision stump is often trained by brute force: discretize the real numbers from the smallest to the largest value in the training set, enumerate all possible classifiers, and pick the one with the lowest training error. One can be more clever in the discretization: between each pair of data points, only one classifier must be tested (since any stump in this range will give the same value).

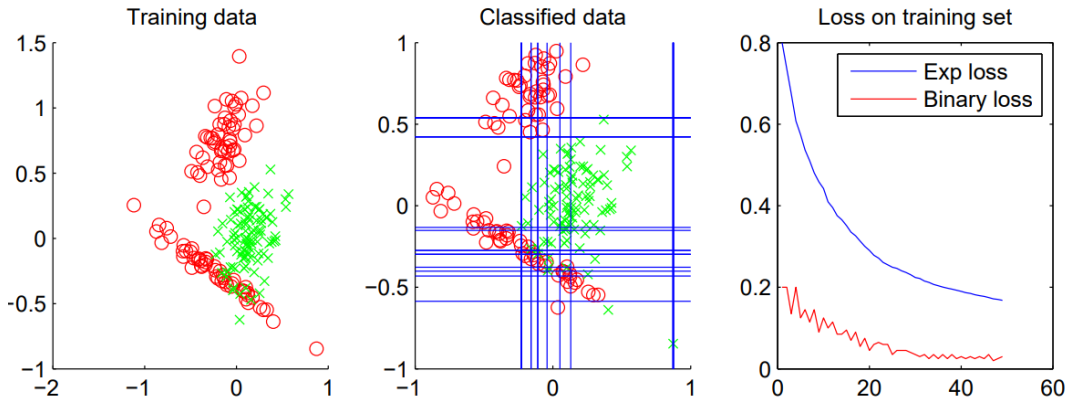


Figure 3.2 AdaBoost classifier uses decision stumps to learn the data.

Firstly, a model is built from the training data. Then the second model is built which tries to correct the errors present in the first model. This procedure is continued and models are added until either the complete training data set is predicted correctly or the maximum number of models is added. We have used two Boosting algorithms in detecting fraudulent trips the AdaBoost and XGBoost Algorithm.

3.4 XGBOOOST ALGORITHM

XGBoost is a decision-tree-based ensemble Machine Learning algorithm that uses a gradient boosting framework. In prediction problems involving unstructured data (images, text, etc.) artificial neural networks tend to outperform all other algorithms or frameworks.

XGBOOST

Data: Dataset

Intialize $f_0(x)$;

For k=1,2,……., M do

Calculate $g_k=\frac{\partial L(y,f)}{\partial f}$;

Calculate $h_{k=}\frac{\partial^2 L(y,f)}{\partial f}$

Determine the dataset by splitting data into subsets

$$A = \frac{1}{2}\left[\frac{G_L^2}{H_L} + \frac{G_R^2}{H_R} - \frac{G^2}{H}\right]$$

Determine the size of each subset $\omega^* = -\frac{G}{H}$

Determine the base learner$b(x) = \sum_{j=1}^{T} \omega I$

Add all subsets$f_k(x) = f_{k-1}(x) + b(x)$

end

Result: f(x)=$\sum_{k=0}^{M} f_k(x)$

- In this algorithm, decision trees are created in sequential form. Weights play an important role in XGBoost. Weights are assigned to all the independent variables which are then fed into the decision tree which predicts results.
- The weight of variables predicted wrong by the tree is increased and these variables are then fed to the second decision tree. These individual classifiers/predictors then ensemble to give a strong and more precise model. It can work on regression, classification, ranking, and user-defined prediction problems.
- The XGBoost algorithm divides the data into a number of subsets and classifies each weak learner, then combines all of the weak learner classifications to build a strong learner. The data is categorised in the model as to whether or not miscellaneous charges are present for a trip, and each subset is then combined to provide the final result. In terms of classification, the model predicts the accuracy.

3.5. TECHNOLOGY USED

- The developed model runs using the Integrated Development and Learning Environment in Python integrated development environment that comes pre-installed with the language's default implementation.
- IDLE contains a full-featured text editor with syntax highlighting, auto-completion, and smart indent for writing Python scripts. There's also a debugger with stepping and breakpoints.
- Multi-window text editor features are syntax highlighting, autocomplete, smart indent, and other features. Stepping, persistent breakpoints, and call stack visibility are all included in the integrated debugger.
- The "Shell Window" and the "Editor Window" are the two most important windows. The "Shell Window" gives you access to interactive Python, while the "Editor Window" lets you create or modify Python files.

3.6 TAXI TRIP DATASET

The chh-ola dataset, which has 17 fields and 10,48,574 data points, was chosen from the Kaggle database to aid in the prediction of taxi fares. The data is in Comma-separated Volume (CSV) format and contains a variety of datatypes.The operation is carried out after extracting the appropriate field from this dataset.

The description of the fields in the dataset is as follows:

- ID-to find the data with the unique identifier
- vendor_id taxi data providing vendor; 1 = TaxiTech Inc. 2 = DataCollectors Inc pickup_loc - Location ID from where passenger was picked up
- drop_loc - Location ID where a passenger was dropped
- driver_tip - Tip given to the driver
- mta_tax - Automatically triggered tax amount
- distance - distance covered in the trip
- pickup_time - Date/Time when meter started
- drop_time - Date/Time when meter stopped
- num_passengers - Cab passenger count
- toll_Amt - Toll paid in the booths
- payment_Method - Method of payment symbolised by a numeric code (1 = Credit Card, 2 = Debit Card, 3 = Cash)
- rate_code - Rate code for the trip (1 = Standard, 2 = Airport, 3 = Connaught Place, 4 = Noida, 5 = Negotiated Fare, 6 = Pooled ride)
- stored_flag - Flag which signifies whether trip data was immediately sent to Chh-OLA's database or not (Y=Yes, N=No, because of connection error)
- extra_charges - Miscellaneous charges(waiting time)
- improvement_charge - Charge levied for improvement in infrastructure

- total_amount - Output label, Final amount to be paid including meter fare and all extra charges.

The dataset's taxi trips are employed to obtain the fraudulent trip using additional costs. Each factor's numerical and decimal values are included in this data. The data in the dataset is prepared in the appropriate data format for the model to be trained. The data in the dataset is filtered throughout the feature selection phase, preparing it for feeding to the model, which will yield the desired output.

3.7. PREPROCESSING

- During preprocessing, the data is imported to transform it to a machine-readable format. The data is translated into a specific numerical format to work in the entire process, and the needed field for testing and training is selected from the dataset.
- The null values in the dataset are removed, and the data is filtered. Unless the option is set to True, the function returns a new DataFrame object, and this technique replaces the existing DataFrame instead. The data is then translated into any acceptable existing column to categorical type that converts one data type to another.
- The fields are chosen based on the requirements for detecting fraud rip from the dataset, which is a value from a group of values in a data frame or dataset that corresponds to a specific row or column.
- The input and output features from the dataset are split and fed to the model in the following stage. The procedure begins by dividing the entire data set into two sets of data to train and test the dataset. here Data is used for training 75% of the time and testing 25% of the time. The AdaBoost and XGBoost algorithms are run on this 25% data, and the confusion matrix is created.
- The confusion matrix consists of two labels they are 0 which refers to taxi fraud not occurred and 1 which refers to taxi fraud that occurred. There are four possible occurrences in the confusion matrix they are

- ✓ 00 regular routes are described as such, implying that no taxi fraud has occurred.
- ✓ 01 typical routes are classified as taxi fraud, indicating that no taxi fraud is expected.
- ✓ 10 taxi fraud is said to be common routes, implying that no taxi fraud is expected.
- ✓ 11 taxi scam is referred to as taxi fraud.

- The AdaBoost classifier processes the data and provides the classified result in the confusion matrix, and the accuracy is known once the initial 25% data is given to the model. The XGBoost classifier analyses the same amount of test data and generates a confusion matrix, from which the accuracy is computed. Performance measures such as precision score, recall score, f1 score, cohen's kappa, and execution time are calculated for both classifiers and presented as a bar graph.
- The ROC (receiver operating characteristic) curve is generated for both the classifier and the performance bar graph to highlight the accuracy of the AdaBoost and XGBoost classifiers. This method identifies the most effective classifier, which is then used to test it.

3.8. FEATURE EXTRACTION

- The relevant data fields are extracted from the dataset and stored as processed data to execute the task of identifying the fraud trip. The best classifier is picked based on the accuracy and evaluation metrics, which is the XGBoost classifier, which produces the result with efficiency and speed. In the conversation window, you can also enter text for the pricing and include a label with editable text with its width and height.
- The user interface includes labels and words in a grid style, as well as a submit button for receiving the output. The module's main goal is to detect

fraud by estimating the price that a taxi driver will charge for a passenger based on extraneous costs.

- The result response looks for similar data in the dataset and returns the results. The taxi fraud detection window consists of the required label in finding the fraud trip they are the pick_up label, Drop label, payment method and the amount asked by the driver and a submit button. The pick-up and drop label consists of ids from 1 to 266 ola dataset consists of these number of location id over India. The payment method has three options credit card, Debit card, and cash. When the value is entered to the user interface panel in applying the best classifier XGBoost, it works faster to give the result.
- Throughout the feature extraction process, extraneous expenses are identified, with a value of 0 indicating no fraud and a value greater than 0 indicating that the driver has committed fraud on the passenger. If there is an additional fee in the total amount when a passenger travels from point A to point B, it is likely that the trip is fraudulent; otherwise, the driver will ask for a genuine fare. This detects a taxi fraud trip and displays it to the passenger so that they may see the fare and negotiate with the driver to avoid being overcharged.

CHAPTER 4

EXPERIMENTAL RESULTS AND PERFORMANCE EVALUATION

In this section, the proposed method has been evaluated with the performance metrics and the results of the system on both real-world and synthetic trajectory data sets.

4.1. EVALUATION METRICS

Confusion Matrix

A confusion matrix is a technique for summarizing the performance of a classification algorithm. Classification accuracy alone can be misleading if you have an unequal number of observations in each class or if you have more than two classes in your dataset. Calculating a confusion matrix can give a better idea of what the classification model is getting right and what types of errors it is making.

Calculation of confusion Matrix.

1. A test dataset or a validation dataset with expected outcome values is taken.
2. a prediction for each row in the test dataset is done
3. From the expected outcomes and predictions count:
    1. The number of correct predictions for each class.
    2. The number of incorrect predictions for each class, organized by the class that was predicted.
    3. Expected down the side: Each row of the matrix corresponds to a predicted class.
    4. Predicted across the top: Each column of the matrix corresponds to an actual class.

The counts of correct and incorrect classification are then filled into the table. The total number of correct predictions for a class goes into the expected row for that class value and the predicted column for that class value. In the same way, the total number of incorrect predictions for a class goes into the expected row for that class value and the predicted column for that class value.

	Positive	Negative
Positive	True positive	False Positive
Negative	False Negative	True Negative

There are four metrics combinations in the confusion matrix, which are as follows:

- True Positive: This combination tells us how many times a model correctly classifies a positive sample as Positive?
- False Negative: This combination tells us how many times a model incorrectly classifies a positive sample as Negative?
- False Positive: This combination tells us how many times a model incorrectly classifies a negative sample as Positive?
- True Negative: This combination tells us how many times a model correctly classifies a negative sample as Negative?

Precision Score

Precision is defined as the ratio of correctly classified positive samples (True Positive) to a total number of classified positive samples (either correctly or incorrectly). precision helps us to visualize the reliability of the machine learning model in classifying the model as positive. While calculating the Precision of a

model, we should consider both Positive as well as Negative samples that are classified. If there is a requirement of classifying all positive as well as Negative samples as Positive, whether they are classified correctly or incorrectly, then use Precision.

Precision = True Positives / (True Positives + False Positives)

Recall Score

The recall is calculated as the ratio between the numbers of Positive samples correctly classified as Positive to the total number of Positive samples. The recall measures the model's ability to detect positive samples. The higher the recall, the more positive samples detected. Unlike Precision, Recall is independent of the number of negative sample classifications. Further, if the model classifies all positive samples as positive, then Recall will be 1. While calculating the Recall of a model, we only need all positive samples while all negative samples will be neglected. if our goal is to detect only all positive samples, then use Recall. Here, we should not care how negative samples are correctly or incorrectly classified the samples.

Recall = True Positives / (True Positives + False Negatives)

F1 Score

If two models have low precision and high recall or vice versa, it is difficult to compare these models. So, for this purpose, we can use F-score. This score helps us to evaluate the recall and precision at the same time. The F-score is maximum if the recall is equal to the precision.

F-Measure = (2 * Precision * Recall) / (Precision + Recall)

Cohen’s Kappa

Cohen’s kappa is a metric often used to assess the agreement between two raters. It can also be used to assess the performance of a classification model. For example, if we had two bankers and we asked both to classify 100 customers in two classes for credit rating (i.e., good and bad) based on their creditworthiness, we could then measure the level of their agreement through Cohen’s kappa. Similarly, in the context of a classification model, we could use Cohen’s kappa to compare the machine learning model predictions with the manually established credit ratings. Like many other evaluation metrics, Cohen’s kappa is calculated based on the confusion matrix. However, in contrast to calculating overall accuracy, Cohen’s kappa takes imbalance in class distribution into account and can, therefore, be more complex to interpret. It tells you how much better your classifier is performing over the performance of a classifier that simply guesses at random according to the frequency of each class. Cohen's kappa is always less than or equal to 1. Values of 0 or less, indicate that the classifier is useless. There is no standardized way to interpret its values.

$$k = (p_o - p_e) / (1 - p_e)$$

p_o: Relative observed agreement among raters

p_e: Hypothetical probability of chance agreement

Accuracy

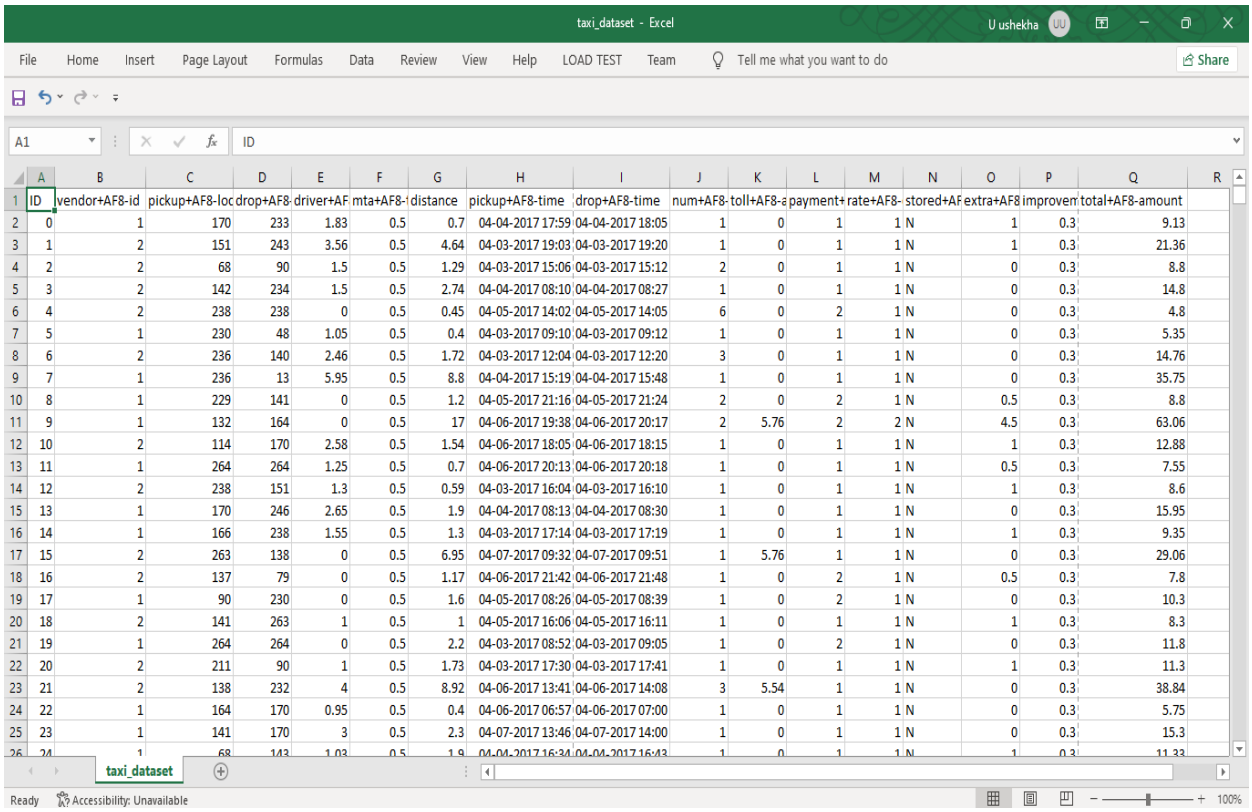
Accuracy is one metric for evaluating classification models. Informally, accuracy is the fraction of predictions our model got right. Accuracy is defined as the percentage of correct predictions for the test data. It can be calculated easily by dividing the number of correct predictions by the number of total predictions.

$$AUC = \frac{(\text{True Positive} + \text{True Negative})}{(\text{True Positive} + \text{True Negative} + \text{False Positive} + \text{False Negative})}$$

ROC Curve

The ROC is a graph displaying a classifier's performance for all possible thresholds. The graph is plotted between the true positive rate (on the Y-axis) and the false Positive rate (on the x-axis)

4.3. EXPERIMENT RESULTS



taxi_dataset - Excel

ID	vendor+AF8-id	pickup+AF8-loc	drop+AF8-	driver+AF	mta+AF8-	distance	pickup+AF8-time	drop+AF8-time	num+AF8-	toll+AF8-a	payment+	rate+AF8-	stored+AF	extra+AF8	improvem	total+AF8-amount
0	1	170	233	1.83	0.5	0.7	04-04-2017 17:59	04-04-2017 18:05	1	0	1	1	N	1	0.3	9.13
1	2	151	243	3.56	0.5	4.64	04-03-2017 19:03	04-03-2017 19:20	1	0	1	1	N	1	0.3	21.36
2	2	68	90	1.5	0.5	1.29	04-03-2017 15:06	04-03-2017 15:12	2	0	1	1	N	0	0.3	8.8
3	2	142	234	1.5	0.5	2.74	04-04-2017 08:10	04-04-2017 08:27	1	0	1	1	N	0	0.3	14.8
4	2	238	238	0	0.5	0.45	04-05-2017 14:02	04-05-2017 14:05	6	0	2	1	N	0	0.3	4.8
5	1	230	48	1.05	0.5	0.4	04-03-2017 09:10	04-03-2017 09:12	1	0	1	1	N	0	0.3	5.35
6	2	236	140	2.46	0.5	1.72	04-03-2017 12:04	04-03-2017 12:20	3	0	1	1	N	0	0.3	14.76
7	1	236	13	5.95	0.5	8.8	04-04-2017 15:19	04-04-2017 15:48	1	0	1	1	N	0	0.3	35.75
8	1	229	141	0	0.5	1.2	04-05-2017 21:16	04-05-2017 21:24	2	0	2	1	N	0.5	0.3	8.8
9	1	132	164	0	0.5	17	04-06-2017 19:38	04-06-2017 20:17	2	5.76	2	2	N	4.5	0.3	63.06
10	2	114	170	2.58	0.5	1.54	04-06-2017 18:05	04-06-2017 18:15	1	0	1	1	N	1	0.3	12.88
11	1	264	264	1.25	0.5	0.7	04-06-2017 20:13	04-06-2017 20:18	1	0	1	1	N	0.5	0.3	7.55
12	2	238	151	1.3	0.5	0.59	04-03-2017 16:04	04-03-2017 16:10	1	0	1	1	N	1	0.3	8.6
13	1	170	246	2.65	0.5	1.9	04-04-2017 08:13	04-04-2017 08:30	1	0	1	1	N	0	0.3	15.95
14	1	166	238	1.55	0.5	1.3	04-03-2017 17:14	04-03-2017 17:19	1	0	1	1	N	1	0.3	9.35
15	2	263	138	0	0.5	6.95	04-07-2017 09:32	04-07-2017 09:51	1	5.76	1	1	N	0	0.3	29.06
16	2	137	79	0	0.5	1.17	04-06-2017 21:42	04-06-2017 21:48	1	0	2	1	N	0.5	0.3	7.8
17	1	90	230	0	0.5	1.6	04-05-2017 08:26	04-05-2017 08:39	1	0	2	1	N	0	0.3	10.3
18	2	141	263	1	0.5	1	04-05-2017 16:06	04-05-2017 16:11	1	0	1	1	N	1	0.3	8.3
19	1	264	264	0	0.5	2.2	04-03-2017 08:52	04-03-2017 09:05	1	0	2	1	N	0	0.3	11.8
20	2	211	90	1	0.5	1.73	04-03-2017 17:30	04-03-2017 17:41	1	0	1	1	N	1	0.3	11.3
21	2	138	232	4	0.5	8.92	04-06-2017 13:41	04-06-2017 14:08	3	5.54	1	1	N	0	0.3	38.84
22	1	164	170	0.95	0.5	0.4	04-06-2017 06:57	04-06-2017 07:00	1	0	1	1	N	0	0.3	5.75
23	1	141	170	3	0.5	2.3	04-07-2017 13:46	04-07-2017 14:00	1	0	1	1	N	0	0.3	15.3

Figure 4.1 Dataset file

The dataset depicted above includes the input data fields selected to train the model to produce the required result. The features from the dataset like pick_up location, Drop Location, extra charges, total charges, etc. The data are imported to the training model with AdaBoost and XGBoost to determine the best classifier.

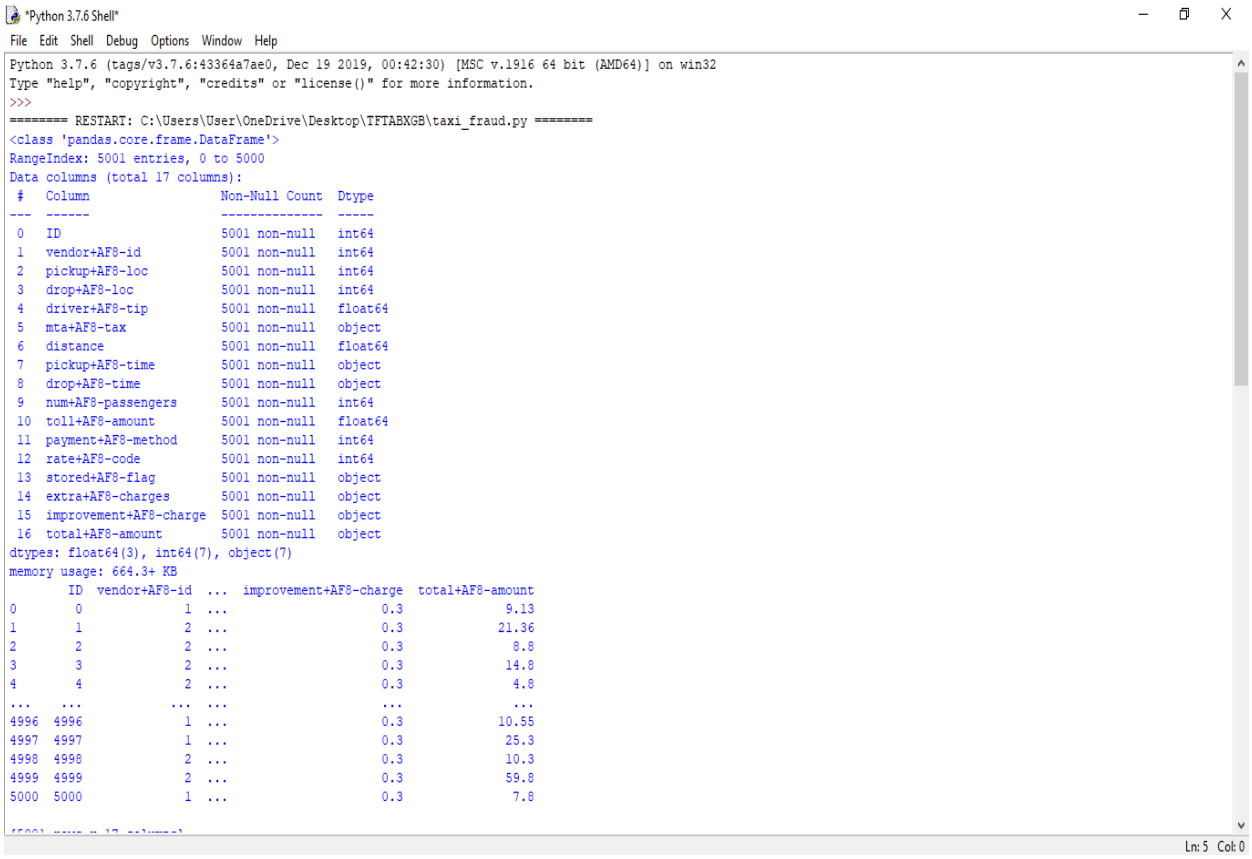


Figure 4.2 Training window

In the training window, whether all the columns have non-null values and the datatype of each field is printed. The feature selection and the data extraction is shown in this window. The prediction is of both classifiers to find the data as fraud or nonfraud using the AdaBoost and XGBoost Machine Learning Algorithm.

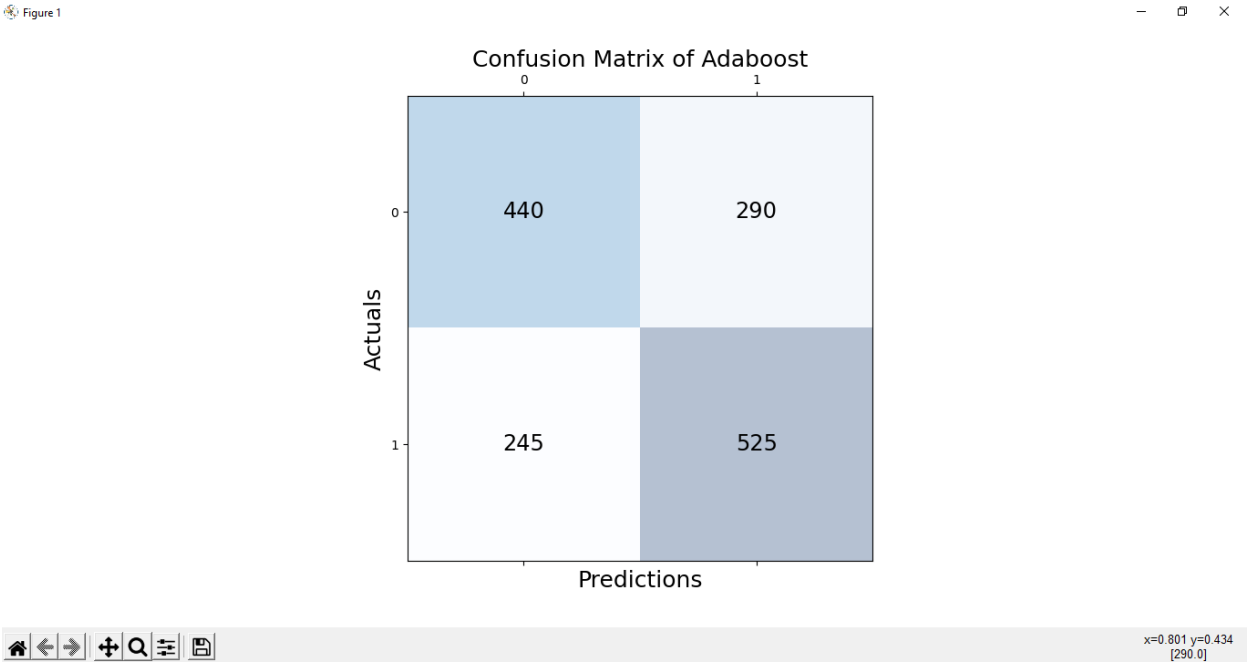


Figure 4.3 Confusion Matrix for AdaBoost

The actual values and the predicted values of the test amount of data are shown in the confusion matrix by the AdaBoost Classifier. Here the Actuals value of 0 represents that the driver didn't overcharge the passenger, and 1 is the passenger has overcharged for the trip.

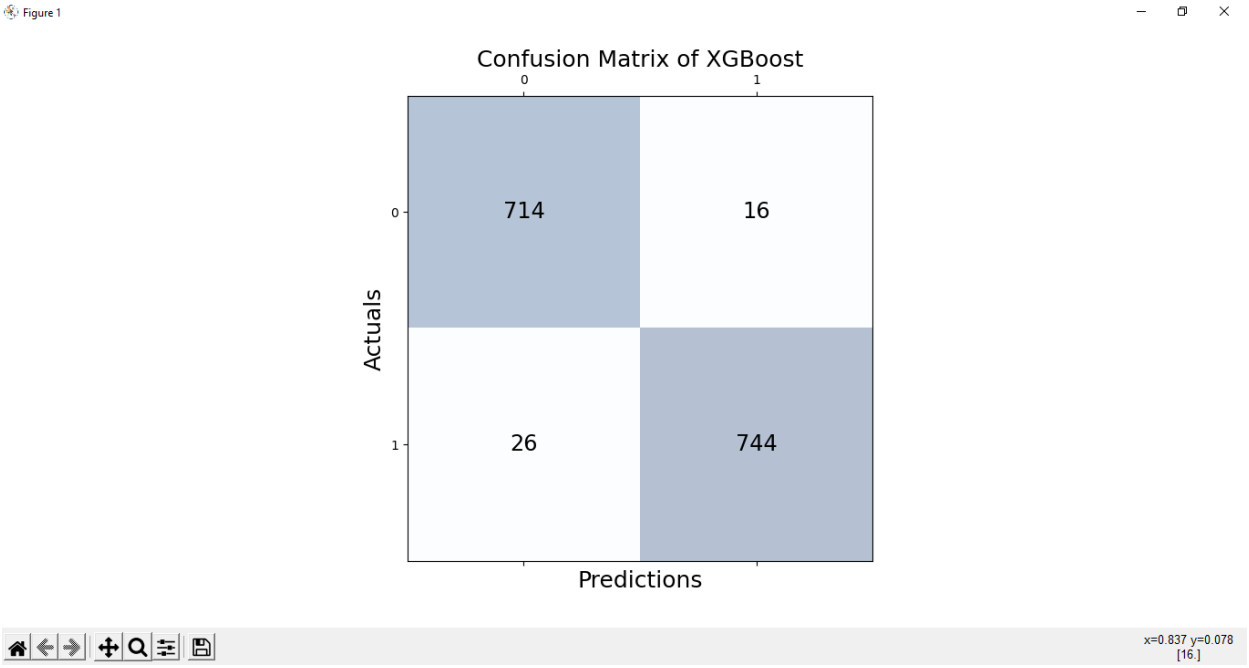


Figure 4.4. Confusion Matrix of XGBoost

The actual values and the predicted values of the test amount of data are shown in the confusion matrix by the XGBoost Classifier.

- ✓ 0,0 represents the number of non-fraud data predicted correctly.
- ✓ 0,1 represents the number of non-frauds that are predicted as fraud.
- ✓ 1,1 represents the number of fraud data predicted correctly.
- ✓ 1,0 represents the number of frauds that are predicted as non-fraud.

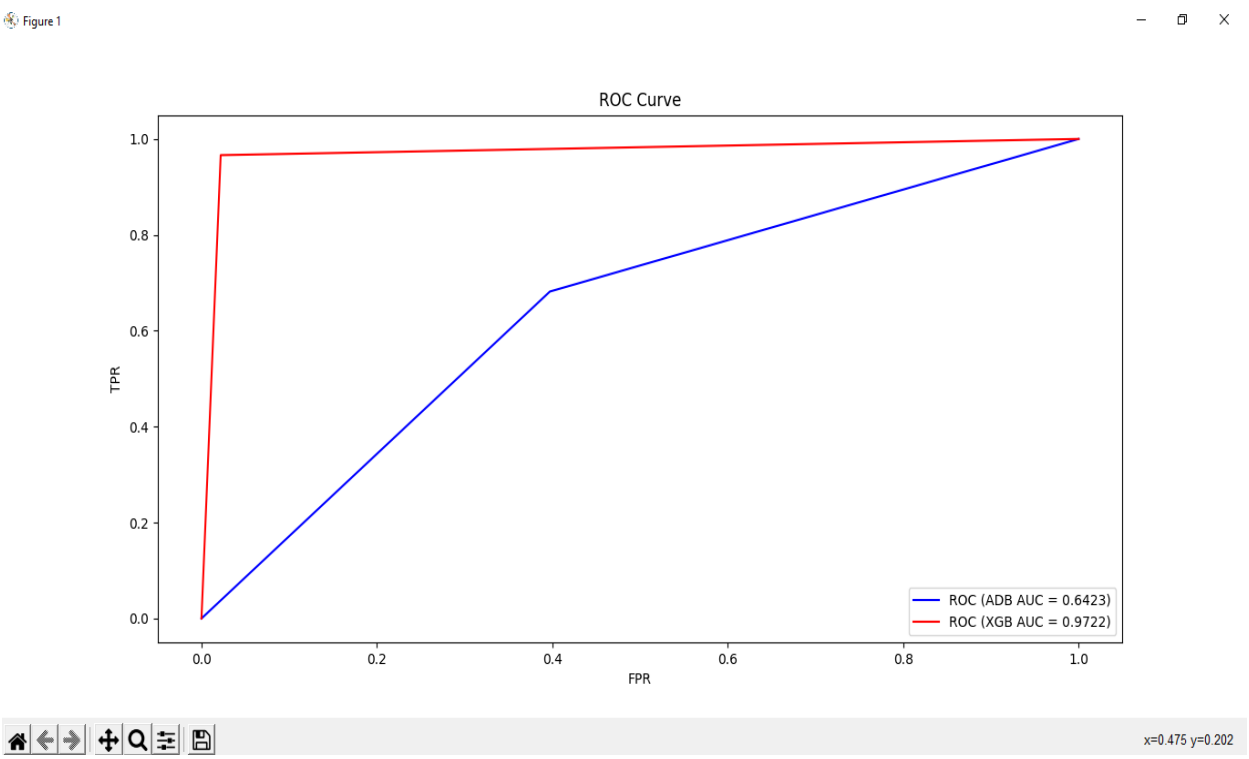


Figure 4.5. ROC Curve for AdaBoost and XGBoost

The curve shows the accuracy of both Algorithm AdaBoost and XGBoost. The ROC curve is drawn to show the performance of both classifier and identify the efficient classifier. It is plotted in terms of True Positive Rate and False Positive Rate values.

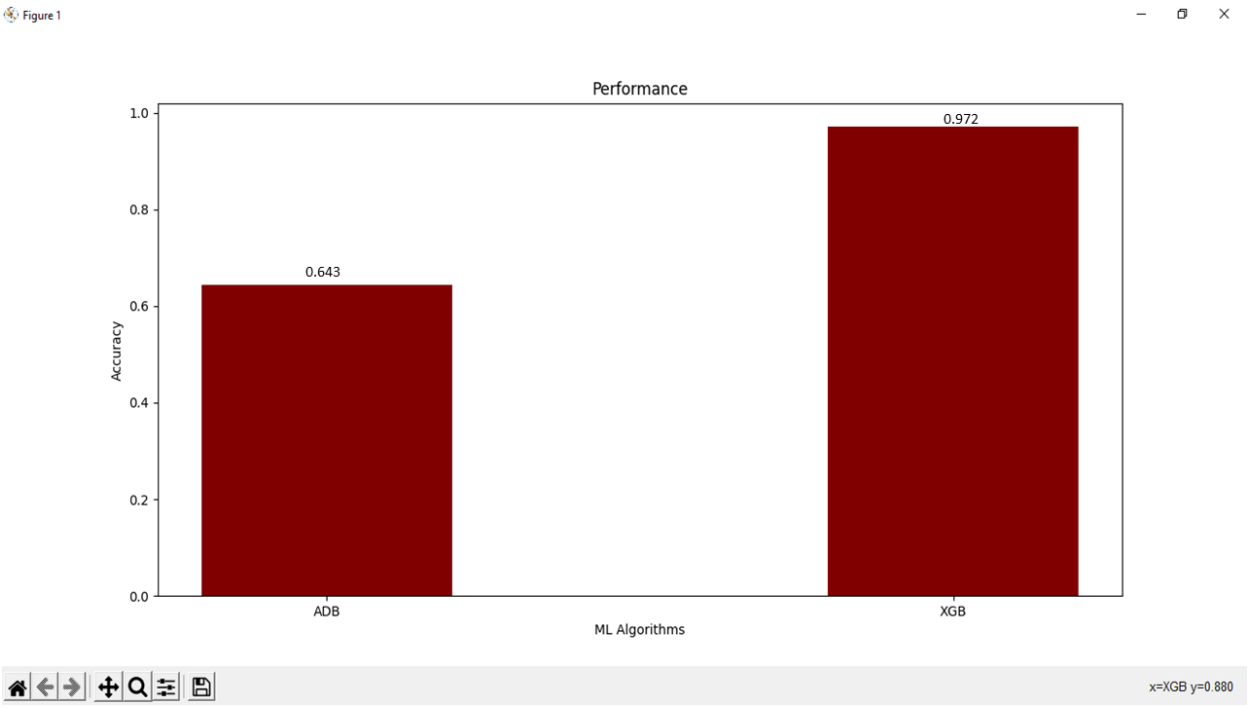


Figure 4.6. The accuracy bar graph in comparison with AdaBoost and XGBoost

The Accuracy is calculated from the Confusion Matrix values that are the true positive, true negative, False positive, and False Negative. The accuracy is calculated for both AdaBoost and XGBoost, Plotted in the Bar graph.

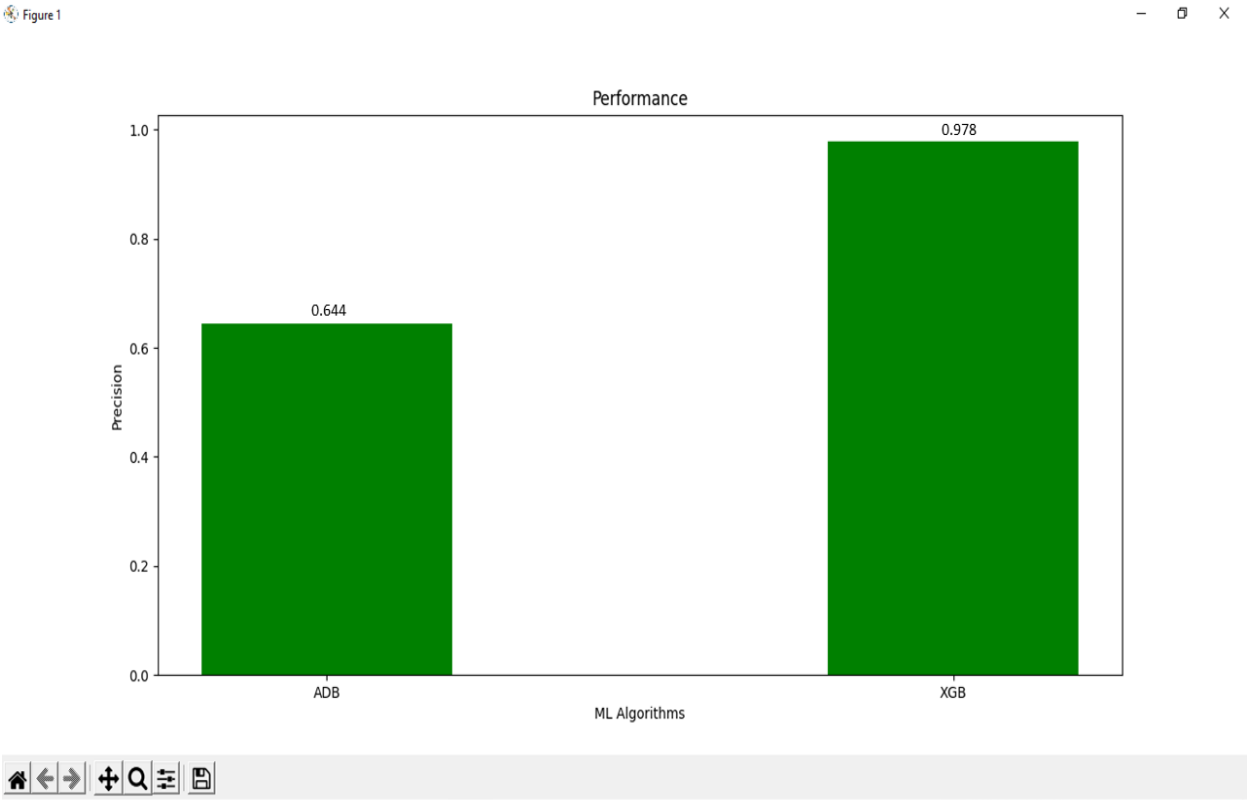


Figure 4.7. Precision bar graph for AdaBoost and XGBoost

Precision is calculated using the true positives and false positives. It shows the ability of a classification model to identify only the relevant data points. The predicted values for AdaBoost and XGBoost are plotted above.

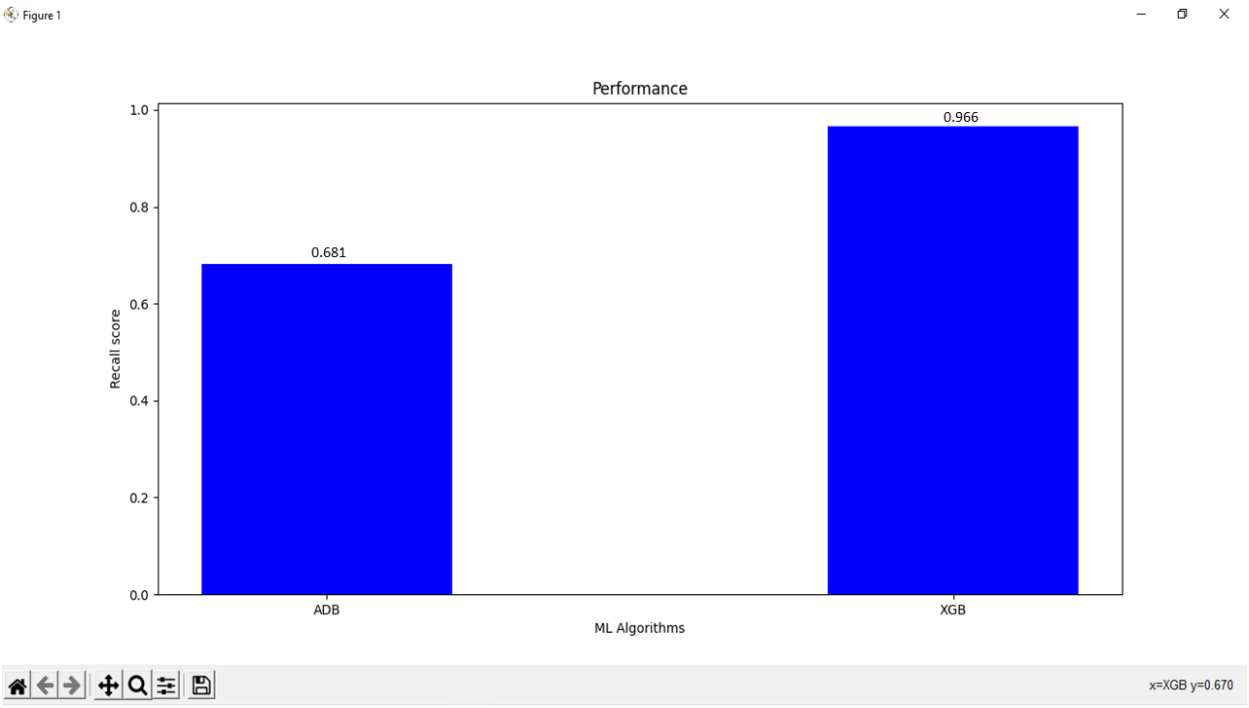


Figure 4.8. Recall Score bar graph for AdaBoost and XGBoost

The recall measures the model's ability to detect a positive sample that is the true positive. It shows the ability of a model to find all the relevant classes with in the dataset. The recall score is calculated for both classifiers and plotted above.

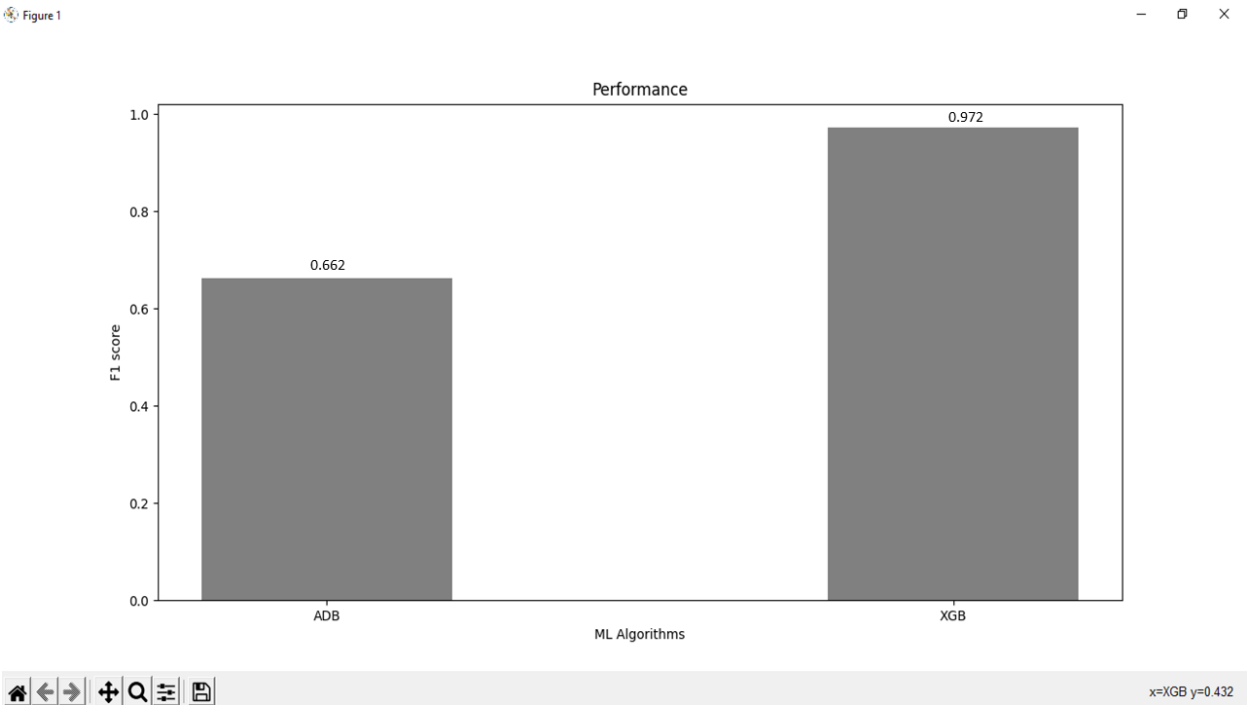


Figure 4.9 F1 Score bar graph for AdaBoost and XGBoost

F1 score is calculated using the values of precision and recall. The f1 score is calculated for both classifiers and is plotted as a bar graph. Here the f1 score of adaboost is 0.62 and the xgboost is 0.972 shows the xgboost has higher f1 score than adaboost.

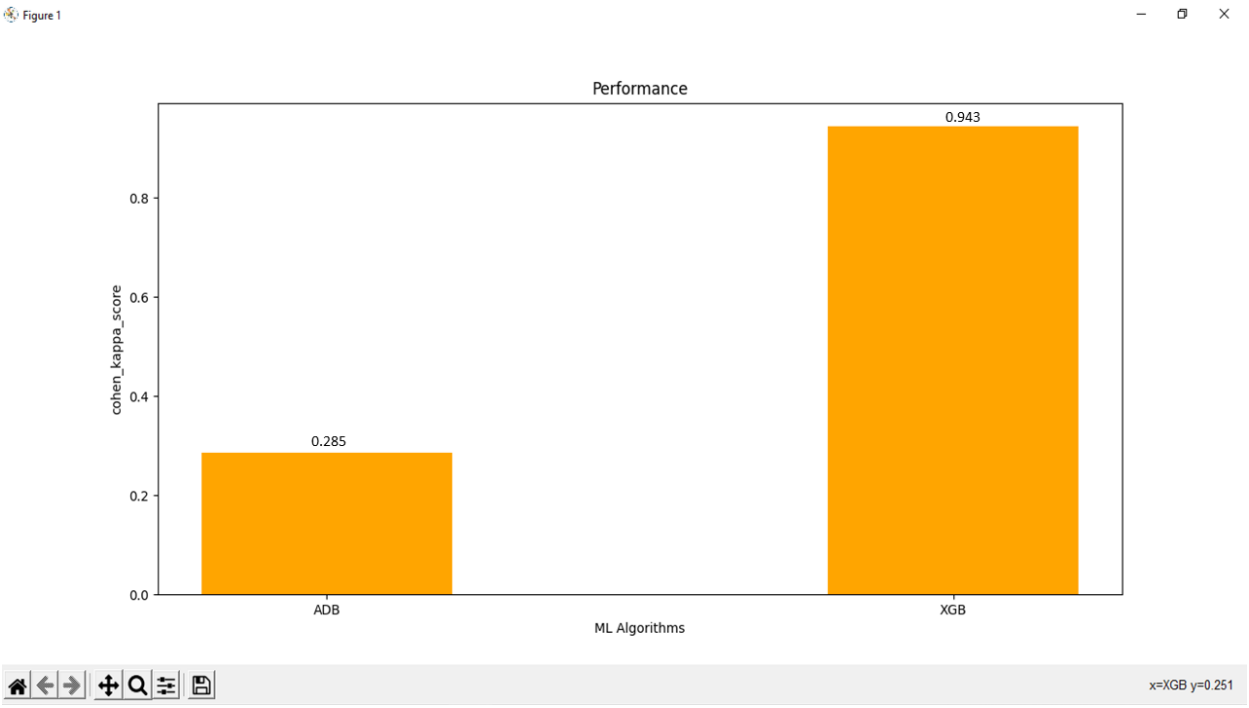


Figure 4.10 Cohen's Kappa Score bar graph for AdaBoost and XGBoost

Cohen's kappa is calculated using the confusion matrix with observed accuracy and the expected accuracy. The Cohen's Kappa value is calculated for both the AdaBoost and XGBoost Classifier and is plotted above.

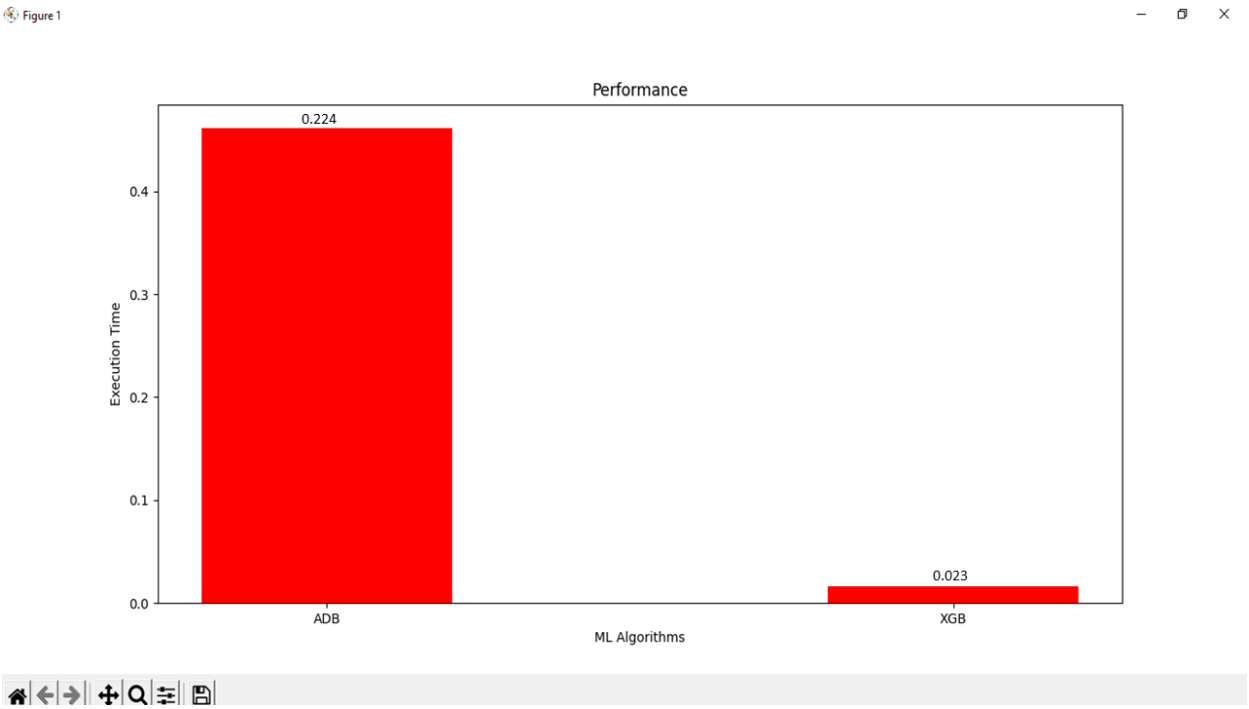


Figure 4.11 Execution time bar graph for AdaBoost and XGBoost

The Execution time for each classifier is taken into account and is plotted as a bar graph above. The execution time of xgboost takes less time in compared to adaboost.

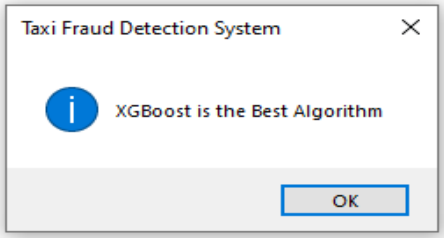


Figure 4.12 Best Classifier Identification

After the Comparison of Performance metrics for both the Algorithm the best Algorithm is found based on the values. The shown result is identified by performing the performance metrics like cohen's kappa, precision, recall, accuracy and f1 score values are compared and comes to conclusion with the classifier.

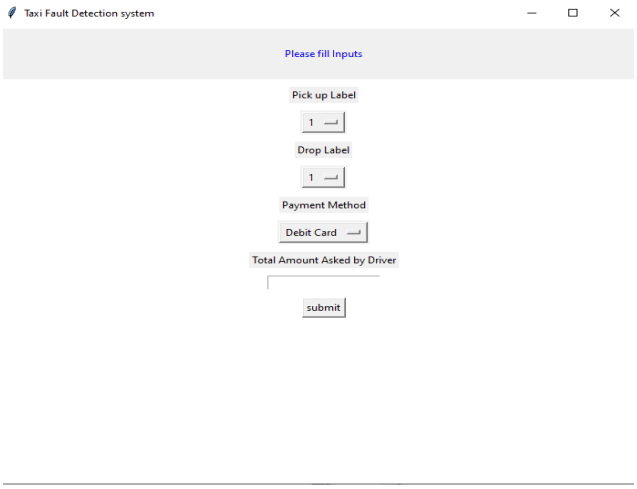


Figure 4.13 User interface Window

This user interface window consists of a pick_up label, Drop label, Payment method, and the total amount asked by the driver.

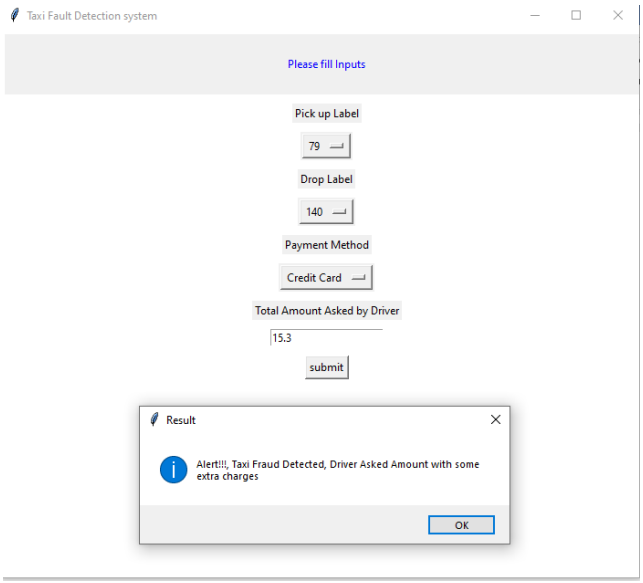


Figure 4.14 Fraudulent trip Detection with user input

The data are given as an input in the user interface window, by the values given by the user for the pick-up, drop-off location and the fare asked by the driver with these data the classifier predicted fraud that the driver charges extraneous charge.

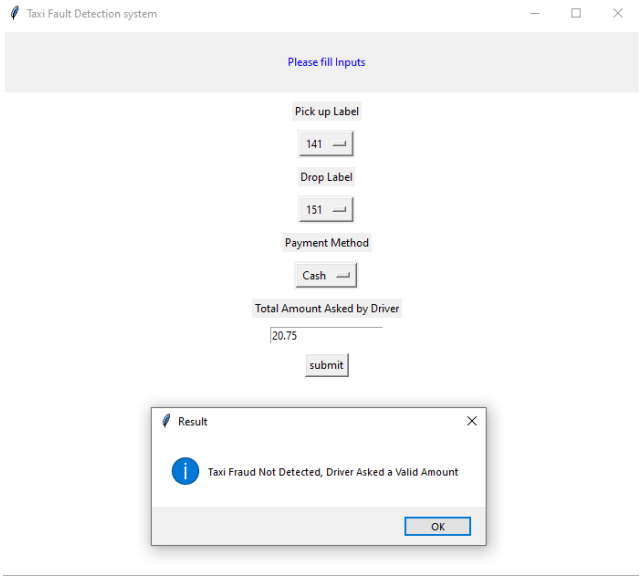


Figure 4.15 User interface with the prediction of Non-Fraud

With the values given as input, the system predicted non-fraud and the amount asked is the valid amount. The pickup location drop-off location is entered, the payment method is picked and the total amount asked by the driver is given as input the classifier checks weather the total amount for that trip has extraneous chargers. In the above figure mentioned trip has no extraneous charger so it is a valid amount charged by the driver.

CHAPTER 5

CONCLUSION

In this project, we considered metered and unmetered taxi trips in real-world data, with the distance and price. To find whether the charges made by the taxi driver are acceptable or not. We proposed the novel system Fiddle Tour: Efficient Trajectory Based Fraudulent Taxi Trip Detection Using Adaboost and Xgboost which predicts the fraud trip by the extra charges in a trip. We have compared both algorithms and identified the best algorithm to improve the efficiency of the system. The Best Classifier is predicted with the help of performance metrics such as accuracy, precision, recall, F1score, Cohen's Kappa, and Execution time. The Classifier predicts whether the taxi driver overcharge or not and it works efficiently. In this system, the efficiency is improved but mapping of the path is not considered it be done in future works.

REFERENCES

1. Y. Ding, W. Zhang, X. Zhou, Q. Liao, Q. Luo, and L. M. Ni, "FraudTrip: Taxi Fraudulent Trip Detection From Corresponding Trajectories," in IEEE Internet of Things Journal, vol. 8, no. 16, pp. 12505-12517, 15 Aug.15, 2021, doi: 10.1109/JIOT.2020.3019398.
2. X. Kong et al., "Spatial-Temporal-Cost Combination Based Taxi Driving Fraud Detection for Collaborative Internet of Vehicles," in IEEE Transactions on Industrial Informatics, vol. 18, no. 5, pp. 3426-3436, May 2022, DOI: 10.1109/TII.2021.3111536.
3. Z. Xiao et al., "On Extracting Regular Travel Behavior of Private Cars Based on Trajectory Data Analysis," in IEEE Transactions on Vehicular Technology, vol. 69, no. 12, pp. 14537-14549, Dec. 2020, DOI: 10.1109/TVT.2020.3043434.
4. S. Liu, L. M. Ni, and R. Krishnan, "Fraud Detection From Taxis' Driving Behaviors," in IEEE Transactions on Vehicular Technology, vol. 63, no. 1, pp. 464-472, Jan. 2014, DOI: 10.1109/TVT.2013.2272792.
5. S. Ma, Y. Zheng and O. Wolfson, "Real-Time City-Scale Taxi Ridesharing," in IEEE Transactions on Knowledge and Data Engineering, vol. 27, no. 7, pp. 1782-1795, 1 July 2015, DOI: 10.1109/TKDE.2014.2334313.
6. M. A. S. Kamal, T. Hayakawa and J. -i. Imura, "Road-Speed Profile for Enhanced Perception of Traffic Conditions in a Partially Connected Vehicle Environment," in IEEE Transactions on Vehicular Technology, vol. 67, no. 8, pp. 6824-6837, Aug. 2018, DOI: 10.1109/TVT.2018.2826067.
7. Z. He, K. Chen and X. Chen, "A Collaborative Method for Route Discovery Using Taxi Drivers' Experience and Preferences," in IEEE Transactions on Intelligent Transportation Systems, vol. 19, no. 8, pp. 2505-2514, Aug. 2018, DOI: 10.1109/TITS.2017.2753468.

8. X. Kong et al., "Mobility Dataset Generation for Vehicular Social Networks Based on Floating Car Data," in IEEE Transactions on Vehicular Technology, vol. 67, no. 5, pp. 3874-3886, May 2018, DOI: 10.1109/TVT.2017.2788441.
9. A. Rossi, G. Barlacchi, M. Bianchini and B. Lepri, "Modelling Taxi Drivers' Behaviour for the Next Destination Prediction," in IEEE Transactions on Intelligent Transportation Systems, vol. 21, no. 7, pp. 2980-2989, July 2020, DOI: 10.1109/TITS.2019.2922002.
10. C. -M. Tseng, S. C. -K. Chau and X. Liu, "Improving Viability of Electric Taxis by Taxi Service Strategy Optimization: A Big Data Study of New York City," in IEEE Transactions on Intelligent Transportation Systems, vol. 20, no. 3, pp. 817-829, March 2019, DOI: 10.1109/TITS.2018.2839265.
11. J. Wang et al., "Anomalous Trajectory Detection and Classification Based on Difference and Intersection Set Distance," in IEEE Transactions on Vehicular Technology, vol. 69, no. 3, pp. 2487-2500, March 2020, DOI: 10.1109/TVT.2020.2967865.
12. Belhadi, Y. Djenouri, G. Srivastava, D. Djenouri, A. Cano and J. C. -W. Lin, "A Two-Phase Anomaly Detection Model for Secure Intelligent Transportation Ride-Hailing Trajectories," in IEEE Transactions on Intelligent Transportation Systems, vol. 22, no. 7, pp. 4496-4506, July 2021, doi: 10.1109/TITS.2020.3022612.
13. Chen *et al.*, "iBOAT: Isolation-Based Online Anomalous Trajectory Detection," in IEEE Transactions on Intelligent Transportation Systems, vol. 14, no. 2, pp. 806-818, June 2013, doi: 10.1109/TITS.2013.2238531.
14. W. Tu et al., "Real-Time Route Recommendations for E-Taxies Leveraging GPS Trajectories," in IEEE Transactions on Industrial Informatics, vol. 17, no. 5, pp. 3133-3142, May 2021, doi: 10.1109/TII.2020.2990206.

15. Y. Lai, Z. Lv, K. -C. Li and M. Liao, "Urban Traffic Coulomb's Law: A New Approach for Taxi Route Recommendation," in IEEE Transactions on Intelligent Transportation Systems, vol. 20, no. 8, pp. 3024-3037, Aug. 2019, doi: 10.1109/TITS.2018.2870990.
16. H. Rong et al., "Mining Efficient Taxi Operation Strategies From Large Scale Geo-Location Data," in IEEE Access, vol. 5, pp. 25623-25634, 2017, doi: 10.1109/ACCESS.2017.2732947.
17. J. Li et al., "A Traffic Prediction Enabled Double Rewarded Value Iteration Network for Route Planning," in IEEE Transactions on Vehicular Technology, vol. 68, no. 5, pp. 4170-4181, May 2019, doi: 10.1109/TVT.2019.2893173.
18. L. Li, S. Wang and F. -Y. Wang, "An Analysis of Taxi Driver's Route Choice Behavior Using the Trace Records," in IEEE Transactions on Computational Social Systems, vol. 5, no. 2, pp. 576-582, June 2018, doi: 10.1109/TCSS.2018.2831285.
19. Y. Lyu, V. C. S. Lee, J. K. Ng, B. Y. Lim, K. Liu, and C. Chen, "Flexi-Sharing: A Flexible and Personalized Taxi-Sharing System," in IEEE Transactions on Vehicular Technology, vol. 68, no. 10, pp. 9399-9413, Oct. 2019, doi: 10.1109/TVT.2019.2932869.
20. X. Kong, M. Li, T. Tang, K. Tian, L. Moreira-Matias and F. Xia, "Shared Subway Shuttle Bus Route Planning Based on Transport Data Analytics," in IEEE Transactions on Automation Science and Engineering, vol. 15, no. 4, pp. 1507-1520, Oct. 2018, doi: 10.1109/TASE.2018.2865494.
21. X. Zhou, Y. Ding, F. Peng, Q. Luo, and L. M. Ni, "Detecting unmetered taxi rides from trajectory data," 2017 IEEE International Conference on Big Data (Big Data), 2017, pp. 530-535, doi: 10.1109/BigData.2017.8257968.
22. M. Qu, H. Zhu, J. Liu, G. Liu, and H. Xiong, "A cost-effective recommender system for taxi drivers," in Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data

mining. ACM, 2014, pp. 45–54, doi.org/10.1145/2623330.2623668.

23. G. Nagy and S. Salhi, “Heuristic algorithms for single and multiple depot vehicle routing problems with pickups and deliveries,” European journal of operational research, vol. 162, no. 1, pp. 126–141, 2005, https://doi.org/10.1016/j.ejor.2002.11.003.

24. J. Yuan, Y. Zheng, C. Zhang, W. Xie, X. Xie, G. Sun, and Y. Huang, “T-drive: driving directions based on taxi trajectories,” in Proceedings of the 18th SIGSPATIAL International conference on advances in geographic Information systems. ACM, 2010, pp.99–108, doi:10.1145/1869790.1869807.

25. J. Yuan, Y. Zheng, L. Zhang, X. Xie, and G. Sun, “Where to find my next passenger,” in Proceedings of the 13th international conference on Ubiquitous computing. ACM, 2011, pp. 109–118, https://doi.org/10.1145/2030112.2030128.

26. J. W. Powell, Y. Huang, F. Bastani, and M. Ji, “Towards reducingtaxicab cruising time using Spatio-temporal profitability maps.” in SSTD.Springer, 2011, pp. 242–260.

27. K. Yamamoto, K. Uesugi, and T. Watanabe, “Adaptive routing of cruising taxis by mutual exchange of pathways,” in International Conference on Knowledge-Based and Intelligent Information and EngineeringSystems. Springer, 2008, pp. 559–566, doi:10.1504/IJKESDP.2010.030466.

28. B. Li et al., "Hunting or waiting? Discovering passenger-finding strategies from a large-scale real-world taxi dataset," 2011 IEEE International Conference on Pervasive Computing and Communications Workshops 2011, pp. 63-68, doi:10.1109/PERCOMW.2011.5766967.

29. J. Xie, Z. Song, Y. Li, Y. Zhang, H. Yu, J. Zhan, Z. Ma, Y. Qiao, J. Zhang, and J. Guo, “A survey on machine learning-based mobile big data analysis:

Challenges and applications,” Wireless Commun. Mob. Comput., vol. 2018, 19 pages, 2018, https://doi.org/10.1155/2018/8738613.

30. J. Li et al., "An End-to-End Load Balancer Based on Deep Learning for Vehicular Network Traffic Control," in IEEE Internet of Things Journal, vol. 6, no. 1, pp. 953-966, Feb. 2019, doi: 10.1109/JIOT.2018.2866435.

31. J. N. Prashker and S. Bekhor, “Route choice models used in the stochastic user equilibrium problem: A review,” Transp. Rev., vol. 24, no. 4, pp. 437–463, 2004, doi:10.1080/0144164042000181707.

32. C. G. Prato, “Route choice modeling: Past, present, and future research directions," J. Choice Model., vol. 2, no. 1, pp. 65–100, 2009, doi:10.1016/S1755-5345(13)70005-8.

33. M. Furuhata, M. Dessouky, F. Ordo´nez, M.-E. Brunet, X. Wang, and ˜ S. Koenig, “Ridesharing: The state-of-the-art and future directions,” Transportation Research Part B: Methodological, vol. 57, pp. 28–46, 2013, doi:10.1016/j.trb.2013.08.012

34. S. Liu, L. M. Ni, and R. Krishnan, "Fraud Detection From Taxis' Driving Behaviors," in IEEE Transactions on Vehicular Technology, vol. 63, no. 1, pp. 464-472, Jan. 2014, doi: 10.1109/TVT.2013.2272792.

35. D. Zhang, N. Li, Z.-H. Zhou, C. Chen, L. Sun, and S. Li, “ibat: detecting anomalous taxi trajectories from GPS traces,” in Proceedings of the 13th international conference on Ubiquitous computing. ACM, 2011, pp. 99–108, doi.org/10.1145/2030112.2030127.

36. S. Zhang and Z. Wang, “Inferring passenger denial behavior of taxi drivers from large-scale taxi traces,” PloS one, vol. 11, no. 11, 2016, doi:10.1371/journal.pone.0165597.

Stay Ahead

BANNARI AMMAN INSTITUTE OF TECHNOLOGY

An Autonomous Institution Affiliated to Anna University - Chennai • Approved by AICTE • Accredited by NAAC with "A+" Grade
SATHYAMANGALAM - 638401 ERODE DISTRICT TAMILNADU INDIA
Ph : 04295-226000/221289 Fax : 04295-226666 E-mail : stayahead@bitsathy.ac.in Web : www.bitsathy.ac.in

INTERNATIONAL VIRTUAL CONFERENCE ON ADVANCES IN DIGITAL TRANSFORMATION, SOFTWARE TECHNOLOGIES AND INTELLIGENT IOT SYSTEMS (ICADSIS)

This is to certify that **Dr./Mr./Ms. USHEKHA.U** has presented the paper entitled **FIDDLE TOUR: FRAUDULENT TAXI TRIP DETECTION USING KNN MACHINE LEARNING ALGORITHM** in the **International Virtual Conference on Advances in Digital Transformation, Software Technologies and intelligent IoT systems (ICADSIS)**, organized by the Department of Information Technology, Bannari Amman Institute of Technology, Sathyamangalam on 20th and 21st May 2022.

Dr. S. Sundaramurthy
Organising Secretary

Dr. S. Daniel Madan Raja
HoD – IT & Convener

Dr. C. Palanisamy
Principal

Certificate No. : 027

Stay Ahead

BANNARI AMMAN INSTITUTE OF TECHNOLOGY

An Autonomous Institution Affiliated to Anna University - Chennai • Approved by AICTE • Accredited by NAAC with "A+" Grade
SATHYAMANGALAM - 638401 ERODE DISTRICT TAMILNADU INDIA
Ph : 04295-226000/221289 Fax : 04295-226666 E-mail : stayahead@bitsathy.ac.in Web : www.bitsathy.ac.in

INTERNATIONAL VIRTUAL CONFERENCE ON ADVANCES IN DIGITAL TRANSFORMATION, SOFTWARE TECHNOLOGIES AND INTELLIGENT IOT SYSTEMS (ICADSIS)

This is to certify that **Dr./Mr./Ms. RAMYA SP** has presented the paper entitled **FIDDLE TOUR: FRAUDULENT TAXI TRIP DETECTION USING KNN MACHINE LEARNING ALGORITHM** in the **International Virtual Conference on Advances in Digital Transformation, Software Technologies and intelligent IoT systems (ICADSIS)**, organized by the Department of Information Technology, Bannari Amman Institute of Technology, Sathyamangalam on 20th and 21st May 2022.

Dr. S. Sundaramurthy
Organising Secretary

Dr. S. Daniel Madan Raja
HoD – IT & Convener

Dr. C. Palanisamy
Principal

Certificate No. : 028

Big Data & Society BDS-22-0253 External Inbox ×

Big Data & Society <onbehalfof@manuscriptcentral.com> 10:26 AM (2 minutes ago)
to me, 1815042, muthukumar

13-Jun-2022

Dear Ms. SP:

Your manuscript entitled "Fiddle Tour: Fraudulent Taxi Trip Detection using KNN Machine Learning Algorithm" has been successfully submitted online and is presently being given full consideration for publication in Big Data & Society.

Your manuscript ID is BDS-22-0253.

Please note that if your paper is accepted for publication the authors will be responsible for paying a one-time article processing charge (APC) as outlined at https://journals.sagepub.com/author-instructions/BDS#ArticleProcessingCharge

You have listed the following individuals as authors of this manuscript:
SP, RAMYA; U, USHEKHA; R, Muthukkumar

Please mention the above manuscript ID in all future correspondence or when calling the office for questions. If there are any changes in your street address or e-mail address, please log in to ScholarOne Manuscripts at https://mc.manuscriptcentral.com/bdas and edit your user information as appropriate.

You can also view the status of your manuscript at any time by checking your Author Center after logging in to https://mc.manuscriptcentral.com/bdas