FIDDLE TOUR: FRAUDULENT TAXI TRIP DETECTION USING KNN MACHINE LEARNING ALGORITHM

MINI PROJECT REPORT (15IT71C)

Submitted by

S.P.RAMYA (1815024)

U.USHEKHA (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

NOVEMBER 2021

ANNA UNIVERSITY: CHENNAI 600 025

BONAFIDE CERTIFICATE

Certified that this project report **“FIDDLE TOUR: FRAUDULENT TAXI TRIP DETECTION USING KNN MACHINE LEARNING ALGORITHM”** is the bonafide work of **S.P. RAMYA (1815024)** and **U. USHEKHA (1815042),** who carried out the project under my supervision.

SIGNATURE	**SIGNATURE**
Dr.K.G.SRINIVASAGAM,M.E., Ph.D,	Dr.R.MUTHUKKUMAR,M.E., Ph.D,
HEAD OF THE DEPARTMENT	**SUPERVISOR**
	Associate professor,
Department of Information Technology	Department of Information Technology
National Engineering College	National Engineering College
K.R. Nagar, Kovilpatti:628503	K.R. Nagar, Kovilpatti:628503
Thoothukudi District, Tamil Nadu.	Thoothukudi District, Tamil Nadu.

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

Internal Examiner **External Examiner**

ABSTRACT

Taxi service is an important part of the public transportation system, providing convenience for our daily life. Taxi services in modern cities are often corrupted by frauds, and passengers are often overcharged by taxi drivers. A passenger is overcharged by the taxi driver is one common type of fraudulent trip, and it brings negative impacts to modern cities. Most existing fraudulent trip detection works rely on the assumption that the trip is correctly recorded by the taximeter. However, there are many taxi drivers in India carrying passengers without activating the taximeter, especially when the taxi driver is trying to overcharge the passengers. Hence the existing system predict the unmetered taxi trips are detected in real-world scenarios, which describes the taxi trip that has been recorded as vacant but has similar driving behaviours to regular metered trips. It consists of a learning model which predicts the occupancy status of taxis, But its prediction level is low and not accurate. In this project we propose a system to detect the taxi fraud by using machine learning algorithm KNN. First, we are going to train our dataset to train the model for fraud detection. Then by testing model the price for the trip is given according to the distance. By this we can detect that the driver cheats with the price or not. By this KNN algorithm we can able to detect taxi fraud at high accuracy.

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.KalidasaMurugavel, M.E., Ph.D.,FIE.,** for providing the necessary facilities in carrying out this project work in our campus.

We sincerely thank to **Dr.K.G.Srinivasagam, 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.,PhD, Associate Professor**, Department of Information Technology, for their valuable guidance. We also thank for their constant encouragement and guidance in bringing out this project as a success.

We express our regards and sincere thanks to faculty coordinators **Dr.B.Paramasivan, M.E,Ph.D., Professor**, **Ms.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

CHAPTER NO.	TITLE	PAGE NO.
	ABSTRACT	iii
	TABLE OF CONTENTS	v
	TABLE OF FIGURES	vi
	TABLE OF ABBREVIATION	vii
1	INTRODUCTION	
	1.1 GENERAL	1
	1.2 MACHINE LEARNING	6
2	LITERATURE REVIEW	
	2.1 LITERATURE REVIEW	8
	2.2 TECHNOLOGY USED	12
	2.3 PACKAGES USED	13
3	PROPOSED SYSTEM	
	3.1 EXISTING SYSTEM	17
	3.2 PROPOSED SYSTEM	20
	3.2.1 MODULES	21
	3.2.2 MODULES DESCRIPTION	21
4	RESULTS	27
5	CONCLUSION	38
6	REFERENCE	39

TABLE OF FIGURES

FIGURE NO.	FIGURES NAME	PAGE NO.
3.a	Proposed System Flow	20
4.1	Dataset file	27
4.2	Training Window	28
4.3	Testing Window	29
4.4	Empty User Interface Window	30
4.5	User Interface Window with AC	31
4.6	Price prediction for AC trip	32
4.7	Price prediction for non-AC trip	33
4.8	Fraud Detection with One way and non-AC	34
4.9	Fraud Detection with One way and AC	35
4.10	Fraud Detection with Round Trip and AC	36
4.11	Fraud Detection with Round Trip and non-AC	37

TABLE OF ABBREVIATION

KNN	**K- Nearest Neighbour**
GPS	**G**lobal **P**ositioning **S**ystem

CHAPTER 1

INTRODUCTION

3.2 **GENERAL**

Taxi services in modern cities are often corrupted and passengers are often overcharged by taxi drivers. Many taxi drivers try to swindle the passengers due to places where difficult to find public transportation services zero hours and prohibited driving after drinking. These taxi frauds result in many complaints and may lead to the bad reputation of taxi services.

Taxi frauds include taximeter tampering, where the taximeter of a taxi is modified so that the shown driving distance is longer than the accurate driving distance. Detour, where a taxi takes an irregular route to deliver a passenger and often takes more driving time and interval than usual. Refusing service, where the driver refuses to carry a particular passenger or tries to find a passenger willing to pay more fare. These fraudulent behaviours usually have evident properties that differ from usual taxi trips.

Unmetered taxi trips are problematic for society because of the quality of the taxi service, competition between taxis, difficulty in tracking where there are no taximeter records for the unmetered rides. The approaches usually rely on the taximeters, which is no longer true for unmetered taxi frauds. There are many other ways to find out whether the taxi is occupied or not, such as the information from seat sensors, but such information is not accurate. that fraud trips exhibit anomalous behaviours from normal metered trajectories, which is also not true for unmetered taxi trips. the driving behaviour or dynamics of unmetered fraud rides are more similar to normal metered trips than to vacant taxi trajectories [1].For tracking the taxi route many other ways are also built that is the detection of outliers that is an object that deviates significantly from the rest of the objects.

taxi frauds focus on discovering individual taxi trajectory frauds. Moreover, there is no work that identifies the change points of taxi trajectory frauds.

The individual and group outliers are identified to find fraud trip in the trajectory dataset. A GPU-based version of the two phase-based algorithm is introduced using the sliding windows strategy to boost the performance and scalability of the process on large scale taxis trajectories. [2].

To find the anomalous detection of the taxi trip the Global Position System(GPS) it becomes a tool in many vehicles for navigation and localization. The traces left behind by GPS-enabled vehicles provide us with an unprecedented window into the dynamics of a city's road network. This information has been analyzed to uncover traffic patterns , city dynamics, driving directions , a city's "hot spots" , finding vacant taxis around a city [31], and good taxi operation patterns. To explore using statistical approaches to enhance detection performance and data processing efficiency. [3]. The GPS technology and new forms of urban geography have changed the paradigm for mobile services. The abundant availability of GPS traces has enabled new ways of doing taxi business, effective route recommendation are given [12]. Previously many techniques were used in detecting taxi fraud. However, previous methods are not suitable for the detection of unmetered taxi trips. The existing approaches relied on the taximeters, which is no longer correct for unmetered taxi frauds. There are many other ways to find out whether the taxi is occupied or not, such as the information from seat sensors, but such information is not accurate[11].Formerly they used Map-matching, which is a calibration technique that aims to align the location points of a trajectory to a road network, thus making the position data accurate for feature extraction[1]. Some previous works on taxi fraud detection try to avoid performing the map-matching task to save pre-processing time, its accuracy is low when compared to Machine Learning Algorithm. An increased amount of effort has been focused on optimizing the selection of routes for taxis, as part of the development of smart urban environments, and the increase of the accumulated trajectory data sets. One challenging issue is to match and recommend appropriate cruising routes to taxis, as most taxis cruise on streets aimlessly looking for passengers. Drivers encounter lots of difficulty in optimizing their cruise routes and hence increasing their incomes, and such inability not only decreases their profit but also increases the traffic load in urban cities. The concept of Urban Traffic Coulomb's law is proposed to model the relationship between taxis and passengers in urban cities. Taxis and passengers are viewed as positive and negative charges. Formulas that calculate the traffic charges and traffic forces are also defined within this concept. Cruising is not the only way to find passengers, and waiting at temporary places, e.g., taxi stops, may be an efficient option compared to cruising around [5].A better strategy not only helps taxi drivers earn more with less effort, but also reduce fuel consumption and carbon emissions. It is interesting to examine the influential factors in passenger seeking strategies and find algorithms to guide taxi drivers to passenger hotspots with the right timing. With the abundant availability of history taxicab traces, the existing methods of doing taxi business have been radically changed. An important metrics such as trip frequency, hot spots and taxi mileage, and provide valuable insights towards more efficient operation strategies. they are in lack of insight into useful operation strategies which can benefit the taxicab companies in the long run. taxicab services can be availed in two different modes: centralized –where a centralized booking office allocates taxicabs to customers, and ad-hoc –where taxicabs can be hailed directly from the road. The major advantage of taxicab services is their universal availability at any given time and location, unlike public transportation, which operates on fixed routes and shuts down during late hours. There are several situations where taxicab services are indispensable, such as medical emergencies, traveling with heavy luggage, traveling with old or physically challenged people, and traveling with infants[6].[7] Effective route planning is the key to improving transportation efficiency. By leveraging the in-depth knowledge of road topology and traffic trends, experienced drivers (e.g., taxi drivers) can usually find near-optimal routes Without comprehensive real-time traffic information, drivers can only make routing decisions based on their limited visions. These short-sighted and non-cooperative routing decisions inevitably deteriorate the resource utilization efficiency of road networks combined with these state-of-the-art technologies, big data and machine learning play an increasingly important role in reducing traffic congestion,[19] improving road safety, and enhancing driving comfort. Understanding travellers' route choice behaviour is a key element in transportation modelling and urban planning. An appropriate route choice model can help explain travellers' perceptions of route characteristics and predict their actions under certain hypothetical scenarios. A traditional approach to defining the route choice process, and the subsequent traffic equilibrium, is to assume a deterministic behaviour. This deterministic equilibrium usually states optimality conditions, such as minimizing transport costs or satisfying Wardrop's first principle of traffic equilibrium . These models assume that travellers have perfect information and seek to unilaterally minimize their travel costs. Typically, a mathematical programming model is formulated and solved by an iterative algorithm. If applied with care and understanding, a deterministic user-equilibrium model provides a simple but effective method of traffic assignment [21].[9] Taxi sharing is a promising approach to reducing energy consumptions, utilizing limited taxi resources efficiently while preserving the interest of individuals. The studies mostly fail to locate a pick-up/drop-off point for each individual passenger in scheduling the sharing route. Besides, they can hardly provide personalized services. Flexi-Sharing to provide flexible and personalized taxi sharing services. It considers the nearby alternative pick-up/drop-off locations and schedules a flexible sharing route with the maximum reduced travel distance by letting passengers walk a short distance. [5] Electric vehicles (EVs), along with autonomous vehicles and connected vehicles [2], are bringing disruptive innovations to urban transportation. Taxies, the most used vehicles in cities, have attracted considerable attention for electrification . Several cities, such as New York, Shenzhen, and Beijing, have launched initiatives to promote the use of EVs in the taxi industry. In comparison to gasoline vehicles, however, EVs still have unignorable disadvantages at present, such as short driving ranges and long charging times. The widespread adoption of ETs is currently facing great challenges, thus highlighting the need for effective policies and strategies to promote ETs. The detour distance in a sharing schedule could be significantly shortened if passengers are willing to walk a short distance before getting on or after getting off the taxi[22].[31] A profitable taxi route recommendation method called adaptive shortest expected cruising route (ASER). In ASER, a probabilistic network model is developed to predict pick-up probability and capacity of each location by using Kalman filtering method. To recommend profitable driving routes to taxi drivers, ASER takes the load balance between passengers and taxis into consideration and the shortest expected cruising distance is introduced to formulate potential cruising distance of taxis. [15] taxi operation management mostly focus on finding optimal driving strategies or routes, lacking in-depth analysis on what the drivers learned during the process and how they affect the performance of the driver. In this work, we make the first attempt to inversely learn the taxi drivers' preferences from data and characterize the dynamics of such preferences over time. In conclusion, we propose a new type of taxi fraud to detect the Taxi trip price fraud using a Machine learning algorithm. In Machine learning, many algorithms are available the KNN algorithm used in our project. We propose the system of "Fiddle tour: Fraudulent Taxi Trip Detection using KNN Machine Learning Algorithm". By using Machine learning, it will be easy to identify taxi fraud. With the KNN algorithm used here, the result accuracy is high. The passenger may know the price for their ride, with Ac/Non Ac, total distance, the place they start their trip by using this application.

1.2 MACHINE LEARNING

Machine learning algorithm is the ability to automatically learn and improve from experience without being explicitly programmed. By using the Machine learning algorithm it will be easy for us to train our dataset with any number of values. In Machine learning we used the KNN algorithm. The KNN(K- Nearest Neighbour) Algorithm is a simple, easy-to-implement supervised machine learning algorithm that can be used to solve both classification and regression problems. It is very simple algorithm to understand and interpret. It is very useful for nonlinear data because there is no assumption about data in this algorithm. KNN works on a principle assuming every data point falling in near to each other is falling in the same class.It is a versatile algorithm as we can use it for classification as well as regression. It has relatively high accuracy, no assumptions about data ,no need to make additional assumptions, tune several parameters, or build a model. This makes it crucial in nonlinear data case. A supervised machine learning algorithm is one that relies on labelled input data to learn a function that produces an appropriate output when given new unlabelled data. A classification problem has a discrete value as its output. A regression problem has a real number (a number with a decimal point) as its output. The KNN algorithm assumes that similar things exist in close proximity. In other words, similar things are near to each other. It is a versatile algorithm also used for imputing missing values and resampling datasets. As the name (K Nearest Neighbour) suggests it considers K Nearest Neighbours (Data points) to predict the class or continuous value for the new Datapoint. For classification a class label assigned to the majority of K Nearest Neighbours from the training dataset is considered as a predicted class for the new data point. For regression the Mean or median of continuous values assigned to K Nearest Neighbours from training dataset is a predicted continuous value for our new data point. In our project KNN is used as a classifier that classifies the input data from the dataset. From that data the output is calculated and displayed by training the model. During the training process of the model first pre-processing is done by tokenization and stemming process. Then features selection from the input given by the passenger and is proceed into bag of words before feeding it to the KNN module. we’ll try to test the model’s accuracy for different K values. The value of K that delivers the best accuracy for both training and testing data is selected. The KNN algorithm is used to train the machine by using dataset to find what will be the response for a specific query that is given by the passenger, that the total distance and price according to the feature selection of the passenger and for a specific path. It will classify the input data to predict the output based the responses provided in the dataset and the final output is displayed. It acts as a learning algorithm for a machine.

CHAPTER 2

LITERATURE REVIEW

2.1 LITERATURE REVIEW

According to Ye Ding, et al. [1] proposed a system, called “FraudTrip”, which detects “unmetered” taxi trips based on a novel fraud detection algorithm and a heuristic maximum fraudulent trajectory construction algorithm. Based on the experiments on both synthetic and real-world trajectory datasets. In this way, 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. Their work is to use machine learning techniques for prediction.

Asma Belhadi, et al. [2] approach allows to identify both individual and group outliers and is based on a two phase-based algorithm. The first phase determines the individual trajectory outliers by computing the distance of each point in each trajectory, whereas the second identifies the group trajectory outliers by exploring the individual trajectory outliers using both feature selection and sliding windows strategies. A parallel version of the algorithm is also proposed using a sliding window-based GPU approach to boost the runtime performance. Extensive experiments have been carried out to thoroughly demonstrate the usefulness of our methodology on both synthetic and real trajectory databases. the GPU approach enables reaching a speed-up of 341 over the sequential algorithm on large synthetic databases. The efficiency of the proposed method to detect both individual and group trajectory outliers on a real-world taxi trajectory database is also demonstrated in comparison with baseline trajectory outlier and group detection algorithms They improve their approach by using deep neural network.

Chao Chen ,et al. [3] proposed an online method that is able to detect anomalous trajectories “on-the-fly” and to identify which parts of the trajectory are responsible for its anomalousness. Furthermore, we perform an in-depth analysis on around 43 800 anomalous trajectories that are detected out from the trajectories of 7600 taxis for a month, revealing that most of the anomalous trips are the result of conscious decisions of greedy taxi drivers to commit fraud. The proposed isolation-based online anomalous trajectory (Iboat) is evaluated through extensive experiments on large-scale taxi data. The iBOAT achieves state-of-the-art performance, with a remarkable performance of the area under a curve.

Wei Tu, et al. [4] proposed a novel real-time route recommendation system for electric taxi (ET) drivers. Taxi travel knowledge, including the probability of picking up passengers and the distribution of destinations, is learned from the raw GPS trajectories. Taxi travel knowledge is learned from raw GPS trajectories of gasoline taxies and used to estimate the ENRs of sequential actions of ET drivers Considering the cascading effect of route decision making, consecutive ET actions are modelled with an action tree. The corresponding expected net revenue is estimated based on the learned knowledge. A prototype online system is developed for providing route recommendations.

Yongxuan Lai, et al. [5] ,proposed the concept of urban traffic Coulomb’s law is coined to model the relationship between taxis and passengers in urban cities, based on which a route recommendation scheme is proposed. Drivers encounter lots of difficulty in optimizing their cruise routes and hence increasing their incomes, and such inability not only decreases their profit but also increases the traffic load in urban cities.With the help of Urban Traffic Coulomb’s Law model explains the taxis and passengers are viewed as positive and negative charges. It first collects useful information such as the density of passengers and taxis from trajectories, then calculates the traffic forces for cruising taxis, based on which taxis are routed to optimal road segments to pick up desired passengers. Different from existing route recommendation methods, the relationship among taxis and passengers are fully taken into account in the proposed algorithm.

Huigui Rong, et al. [6] approach presents generic insights into the dynamics of taxicab services with the objective of maximizing the profit margins for the concerned parties. It is interesting to examine the influential factors in passenger seeking strategies and find algorithms to guide taxi drivers to passenger hotspots with the right timing. With the abundant availability of history taxicab traces, the existing methods of doing taxi business have been radically changed. We propose important metrics such as trip frequency, hot spots and taxi mileage, and provide valuable insights towards more efficient operation strategies. We analyze these metrics using techniques like Newton’s polynomial interpolation and Gamma distribution to understand their dynamics.

Jinglin Li, et al [7] proposed a double rewarded value iteration network (VIN) to fully learn the experienced drivers’ routing decisions which are based on their implicitly estimated traffic trends. Though comprehensive real-time traffic information can be obtained, route planning remains challenging. Vehicles that plan the fastest/shortest routes by only considering current traffic conditions may fall into newly emerged congestions. First, the global traffic status and routing actions are chronologically extracted from large-scale taxicab trajectories. Then, to model the knowledge of traffic trends, a long short-term memory (LSTM) network is trained. Being expert at learning long-term planning involved functions, the VIN is utilized to model the policy function from both current and predicted future traffic status to an experienced driver's routing action.

Li Li, et al. [8] proposed the route choices of Beijing taxi drivers regarding four frequently mentioned cost-based route choice rules: pursuing shortest time, or distance, avoiding passing signalized intersections, or making left/right turnings. The first study of route choice behaviour goes back to Wardrop's. Wardrop assumed that drivers pursued routes with least costs and proposed the Wardrop's equilibrium in which drivers cannot reduce their travel costs by unilaterally choosing another route. Test results show that route choices of drivers are not always optimal according to either of these rules. Instead, we argue that taxi drivers are bounded rational and usually choose a satisfactory route that belongs to one of the few routes that consume the shortest times.

Yan Lyu, et al. [9] proposed a new taxi-sharing system called Flexi-Sharing to provide flexible and personalized taxi sharing services. Note that only a few studies introduce meeting points to reduce detour distance [8]–[14]. However, they only consider the simplest route structure where passengers meet at one pickup point for getting on a taxi and/or get off the taxi at one drop-off point. It considers the nearby alternative pick-up/drop-off locations and schedules a flexible sharing route with the maximum reduced travel distance by letting passengers walk a short distance. For a sharing request, Flexi-Sharing generates the sharing schedule consisting of a set of companions, the shortest sharing route and the best pickup/drop-off locations by maximizing the satisfaction of involved passengers is maximized.

Xiangjie Kong, et al. [10] proposed a two-stage approach (SubBus), which is composed of travel requirement prediction and dynamic routes planning, based on various crowdsourced shared bus data to generate dynamic routes for shared buses in the “last mile” scene. First, we analyze the resident travel behaviors to obtain five predictive features, such as flow, time, week, location, and bus, and utilize them to predict travel requirements accurately based on a machine learning model. Second, we design a dynamic programming algorithm to generate dynamic, optimal routes with fixed destinations for multiple operating buses utilizing prediction results based on operating characteristics of shared buses. SubBus outperforms other methods on dynamic route planning for the “last mile” scene. Shared bus operation routes at such scenes are usually aimed at trips with fixed destinations.

2.2 TECHNOLOGY USED

IDLE (Integrated Development and Learning Environment) is a Python integrated development environment that comes pre-installed with the language's default implementation. Many Linux distributions include it as an optional element of the Python package. It's written entirely in Python. IDLE, like Python Shell, can be used to execute a single statement as well as develop, modify, and run Python scripts. 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. Its main features, according to the README, are: Multi-window text editor that includes syntax highlighting, autocomplete, smart indent, and other features.Syntax highlighting in the Python shell.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.

2.3 PACKAGES USED

NUMPY

NumPy is the fundamental package for scientific computing in Python. It is a Python library that provides a multidimensional array object, various derived objects (such as masked arrays and matrices), and an assortment of routines for fast operations on arrays, including mathematical, logical, shape manipulation, sorting, selecting, I/O, discrete Fourier transforms, basic linear algebra, basic statistical operations, random simulation and much more.

There are several important differences between NumPy arrays and the standard Python sequences: NumPy arrays have a fixed size at creation, unlike Python lists (which can grow dynamically). Changing the size of an ndarray will create a new array and delete the original. The elements in a NumPy array are all required to be of the same data type, and thus will be the same size in memory. The exception: one can have arrays of (Python, including NumPy) objects, thereby allowing for arrays of different sized elements. NumPy arrays facilitate advanced mathematical and other types of operations on large numbers of data. Typically, such operations are executed more efficiently and with less code than is possible using Python's built-in sequences.

NLTK

NLTK (Natural Language Toolkit) Library is a suite that contains libraries and programs for statistical language processing. It is one of the most powerful NLP libraries, which contains packages to make machines understand human language and reply to it with an appropriate response. Natural Language Processing (NLP) is a process of manipulating or understanding the text or speech by any software or machine. An analogy is that humans interact and understand each other's views and respond with the appropriate answer. In NLP, this interaction, understanding, and response are made by a computer instead of a human.

SCIKIT-LEARN

Scikit-learn (Sklearn) is the most useful and robust library for machine learning in Python. It provides a selection of efficient tools for machine learning and statistical modeling including classification, regression, clustering and dimensionality reduction via a consistence interface in Python. This library, which is largely written in Python, is built upon NumPy, SciPy and Matplotlib. It is used to extract the features from data to define the attributes in image and text data. It is used to identify useful attributes to create supervised models. It is used to check the accuracy of supervised models on unseen data.

TENSORFLOW

Tensorflow is an end-to-end open source platform for machine learning. It has a comprehensive, flexible ecosystem of tools, libraries, and community resources that lets researchers push the state-of-the-art in ML and developers easily build and deploy Machine Learning-powered applications. TensorFlow was originally developed by researchers and engineers working on the Google Brain team within Google's Machine Intelligence Research organization to conduct machine learning and deep neural networks research. The system is general enough to be applicable in a wide variety of other domains, as well. TensorFlow provides stable python and C++ APIs, as well as non-guaranteed backward compatible API for other languages.

TFLearn

TFlearn is a modular and transparent deep learning library built on top of Tensorflow. It was designed to provide a higher-level API to TensorFlow in order to facilitate and speed-up experimentations, while remaining fully transparent and compatible with it. TFLearn features include: Easy-to-use and understand high-level API for implementing deep neural networks, with tutorial and examples. Fast prototyping through highly modular built-in neural network layers, regularizes, optimizers, metrics. Full transparency over Tensorflow. All functions are built over tensors and can be used independently of TFLearn. Powerful helper functions to train any TensorFlow graph, with support of multiple inputs, outputs and optimizers.

TKINTER

In Python, Tkinter is a standard GUI (graphical user interface) package. Tkinter **is** Python's default GUI module and also the most common way that is used for GUI programming in Python. Note that Tkinter is a set of wrappers that implement the Tk widgets as Python classes. Tkinter in Python helps in creating GUI Applications with a minimum hassle. Among various GUI Frameworks, Tkinter is the only framework that is built-in into Python's Standard Library. An important feature in favor of Tkinter is that it is **cross-**platform, so the same code can easily work on Windows**,** macOS**,** and Linux. Tkinter is a lightweight module and simple to use.

KNeighbourClassifier

The K in the name of this classifier represents the k nearest neighbors, where k is an integer value specified by the user. Hence as the name suggests, this classifier implements learning based on the k nearest neighbors. The choice of the value of k is dependent on data. To use this KNeighbourClassifier First, import the KNeighborsClassifier module and create KNN classifier object by passing argument number of neighbors in KNeighborsClassifier() function. Then, fit your model on the train **set** using fit() and perform prediction on the test set using predict() these functions are used.

CHAPTER 3

PROPOSED SYSTEM

3.1 EXISTING SYSTEM

Many algorithms and techniques where implemented to detect the fraud in taxi trip. Ye Ding, et al [1] proposed “FraudTrip”, which detects “unmetered” taxi trips based on a novel fraud detection algorithm and a heuristic maximum fraudulent trajectory construction algorithm. Based on observation, there are 69% unmetered tracking records, and 5% of them are occupied.Li Li, et al. [8] the route choices of Beijing taxi drivers regarding four frequently mentioned cost-based route choice rules: pursuing shortest time, or distance, avoiding passing signalized intersections, or making left/right turnings. Test results show that more than 90% observed traces can be explained under this assumption. This indicates that, though taxi drivers are with rich experience and familiar with the road network, they are with bounded rationality in route choices. Wei Tu, et al. [4] proposed a novel real-time route recommendation system for electric taxi (ET) drivers. The average result for ET that adopt our system still outperforms 50% of real-world gasoline taxi drivers. Leveraging massive-scale taxi GPS trajectory data, this study presents a comprehensive real-time route recommendation system for ET drivers that integrates the cruising on the road and the recharging at stations decision. The average daily net revenue of ET drivers using the developed system outperforms 76.2% of gasoline taxi drivers. Meng Qu , et al[12]the classical Vehicle Routing Problem (VRP), where customers may both receive and send goods. We do not make the assumption common in the VRPPD literature, that goods may only be picked up after all deliveries have been completed. They did not considered traffic at many places. Yan Lyu, et al[9]Taxi sharing is a promising approach to reducing energy consumptions, utilizing limited taxi resources efficiently while preserving the interest of individuals. For a sharing request, Flexi-Sharing generates the sharing schedule consisting of a set of companions, the shortest sharing route and the best pickup/drop-off locations by maximizing the satisfaction of involved passengers. This Flexi sharing Takes many processing time. Asma Belhadi, et al [2]Approach allows to identify both individual and group outliers and is based on a two phase-based algorithm. A parallel version of the algorithm is also proposed using a sliding window-based GPU approach to boost the runtime performance. Extensive experiments have been carried out to thoroughly demonstrate the usefulness of our methodology on both synthetic and real trajectory databases. Experimental results reveal the scalability of the parallel approach compared to the sequential version by reaching a speed-up of up to 341 when dealing with 50, 000 trajectory database When it comes to complex routes they are less accuracy.[10]In the two-stage approach (SubBus) the consider the characteristics of the shared bus route planning and provide a high capacity, short operating distance, and dynamic route planning methods and it can provide effective suggestions for shared bus dynamic route planning, especially from the aspects of operating distance and passengers' number. the predicted accuracy at several stations can reach 80%. Based on the candidate origin set and candidate route set that generated, they obtain the optimal routes for shared buses by our designed dynamic programming algorithm.[15] the first attempt to employ inverse reinforcement learning to analyze the preferences of taxi drivers when making sequences of decisions to look for passengers. The preferences to habits features to gain more knowledge in the learning phase and keep the preferences to profile features stable over time. Daqing Zhang, et al[24] to discover anomalous driving patterns from taxi's GPS traces, targeting applications like automatically detecting taxi driving frauds or road network change in modern cites. firstly grouping of all the taxi trajectories crossing the same source destination cell-pair and represent each taxi trajectory as a sequence of symbols. Secondly, we propose an Isolation Based Anomalous Trajectory (iBAT) detection method and verify with large scale taxi data that iBAT achieves remarkable performance. Detecting over 90% of anomalous trajectories at the false alarm rate of less than 2%.[25] For interactions between stranger to stranger, taxi industry provide fruitful phenomina and evidence to investigate the action decisions. demonstration the big data analytics application in revealing novel insights from massive taxi trace data, which, for the first time, validates the passengers denial in taxi industry. the denial rate of high income taxis is approximately 8.52% passengers are estimated to be denied. Chao Chen, et al [3] Many passengers are victims of fraud caused by greedy taxi drivers who overcharge passengers by deliberately taking unnecessary detours . The detection of these fraudulent behaviors is essential to ensure high-quality taxi service. These frauds are currently detected by manual inspection from experienced staff, based on complaints from passengers. This is rather costly and not very effective as most frauds are not even noticed by passengers if they are unfamiliar with the city. Given that anomalous traces usually deviate significantly from “nor mal” traces, it is possible to automatically detect them by com paring them against a large collection of historical trajectories. To classifying completed trip trajectories as anomalous or normal, iBOAT can work with ongoing trajectories and can determine which parts of a trajectory are responsible for its anomalousness. We validated iBOAT on a large data set of taxi GPS trajectories recorded over a month and found that our method achieved excellent performance (AUC ≥ 0.99 for all data sets), which is comparable to iBAT’s performance.

3.2 PROPOSED SYSTEM

In our work, we have implemented KNN algorithm for our dataset to detect taxi fraud. The dataset provides the features of input and the responses for them. In this project, At first the data's are imported from the input dataset by using JSON library. Then we pre-process the data by tokenization and stemming process that make our data in the data set to get classified int categories and is changed to numerical values by the bag of words to train with the KNN algorithm, then we make feature selection by selecting input features and convert it in to bag of words for loading it in the KNN module. We design a KNN model which can able to give high accuracy, the extracted features are inserted in to the KNN model and the machine gets trained. After training we predict Taxi fraud by feeding test data's into the our application.

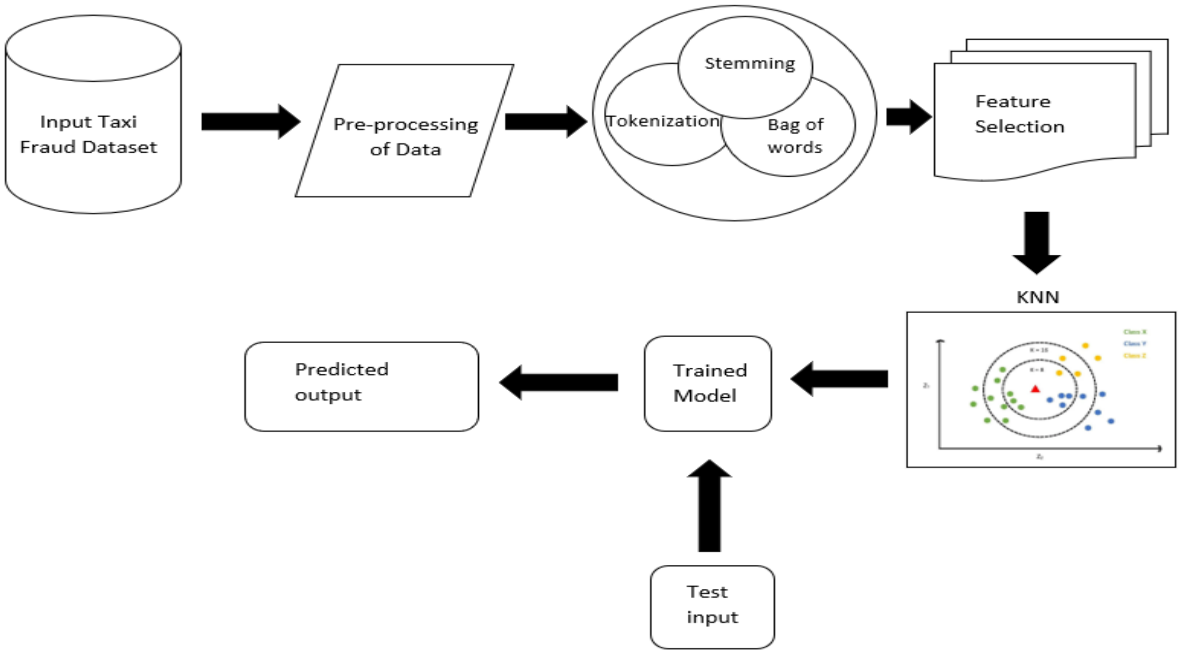


Fig 3.a Proposed System Flow

3.2.1 MODULES

- Dataset
- Training
- Testing

3.2.2 MODULES DESCRIPTION

Dataset(taxi. jason)

The dataset taxi.jason consists of data that the input fields given by the passenger and the responses for the input is produced . This dataset file stores the data of human readable text format to process the data that is JSON files are lightweight, text-based, human-readable, and can be edited using a text editor.. JSON (JavaScript Object Notation) is an open standard file format for sharing data that uses human-readable text to store and transmit data. JSON files are stored with the .json extension. JSON requires less formatting and is a good alternative for XML. JSON is derived from JavaScript but is a language-independent data format. The generation and parsing of JSON is supported by many modern programming languages. application/json is the media type used for JSON. JSON data is written in key/value pairs. The key and value are separated by a colon(:) in the middle with the key on the left and the value on the right. Different key/value pairs are separated by a comma(,).JSON uses less data overall, so you reduce the cost and increase the parsing speed.Readable: The JSON structure is straightforward and readable. You have an easier time mapping to domain objects, no matter what programming language you're working with.The data's are stored in array of huge text data, more than 60 number of data's are there. The intents array stores all the data in the dataset. The fields in the dataset are the from the place were passenger board, to the destination of the passenger, through which way the passenger reaches the destination. The data stored in tag array are the input data given by the passenger, the patterns array given the values of machine understandable details. The tag and patterns provide the fields like the from, to and through. The responses stores the total kilometres for the distance from the boarding place to the destination and the price per kilometre is produced. The feature selection and pre processing are handled by this dataset.

Training(training.py)

In the training phase to train the dataset first to convert the data into machine readable format for this purpose many packages are imported. The packages like NLTK(Natural Language Toolkit) used to build python programs that work with human language data for applying in statistical natural language processing (NLP). It contains text processing libraries for tokenization, parsing, classification, stemming, tagging and semantic reasoning. We use tokenization process and stemming in our project. Tokenization basically refers to splitting up a larger body of text into smaller lines, words or even creating words for a non-English language. Tokenization is one of the most common tasks when it comes to working with text data. Tokenization is essentially splitting a phrase, sentence, paragraph, or an entire text document into smaller units, such as individual words or terms. Each of these smaller units are called tokens. The tokens could be words, numbers or punctuation marks. In tokenization, smaller units are created by locating word boundaries. These are the ending point of a word and the beginning of the next word. These tokens are considered as a first step for stemming and lemmatization. By using these tokens the words in our dataset gets separated. We use K-NN classifier to the training data. To do this we will import the KNeighborsClassifier class of Sklearn Neighbors library. The NumPy package is used for scientific notation in python, we use numpy for processing the array data in the dataset. Tflearn library used to handle machine learning works in our project. Importing os package is for using external file for training that is the jason dataset file, importing jason package is to import jason file to the training. Sklearn package is used to import knn package to the training module. After importing all the packages into the training module the data get trained step by step for the test process. By applying tokenization process the places specified in the dataset each are separated by piece of words. Then the tokenized words are stored to another array for further process, in this way all the data in the dataset are separated and stored. Next these data are go through the stemming process, the separated words are further classified and the root word for places are separated and stored into another array. Stemming is the process of producing morphological variants of a root/base word. Stemming programs are commonly referred to as stemming algorithms or stemmers. Stemming is used in information retrieval systems like search engines. It is used to determine domain vocabularies in domain analysis. Stemming is desirable as it may reduce redundancy as most of the time the word stem and their inflected/derived words mean the same. Now these arrays are sorted and stored into labels array. KNN algorithms use data and classify new data points based on similarity measures. The KNN algorithm only handles numerical data so we need to convert the words to numerical data. We use bag of words to convert the words in form of binary data 0’s and 1’s, each words are classified as 0 or 1 and is stored in a new array. A bag-of-words model, or BoW for short, is a way of extracting features from text for use in modelling, such as with machine learning algorithms. The approach is very simple and flexible, and can be used in a myriad of ways for extracting features from documents. A bag-of-words is a representation of text that describes the occurrence of words within a document. It involves two things: A vocabulary of known words and A measure of the presence of known words.

It is called a bag of words, because any information about the order or structure of words in the document is discarded. The model is only concerned with whether known words occur in the document, not where in the document. The bag-of-words can be as simple or complex as you like. The complexity comes both in deciding how to design the vocabulary of known words (or tokens) and how to score the presence of known words. Another array for overall count of the data present in the dataset. Now these data's are collected and dumped into an empty pickle file for training the data's with the KNN algorithm. The KNN algorithm is used to train the machine by using dataset to find what will be the response for a specific query. When the training process is done two files will be created ,one is the pickle file that stores all the values and the knn pickle file which is trained by the dataset. Once this file is created we need not train the model again. The whole dataset will be trained by this module.

Testing(test1.py)

By using the KNN algorithm once the dataset is trained the machine will understand the data easily and executes faster to provide the result within seconds.

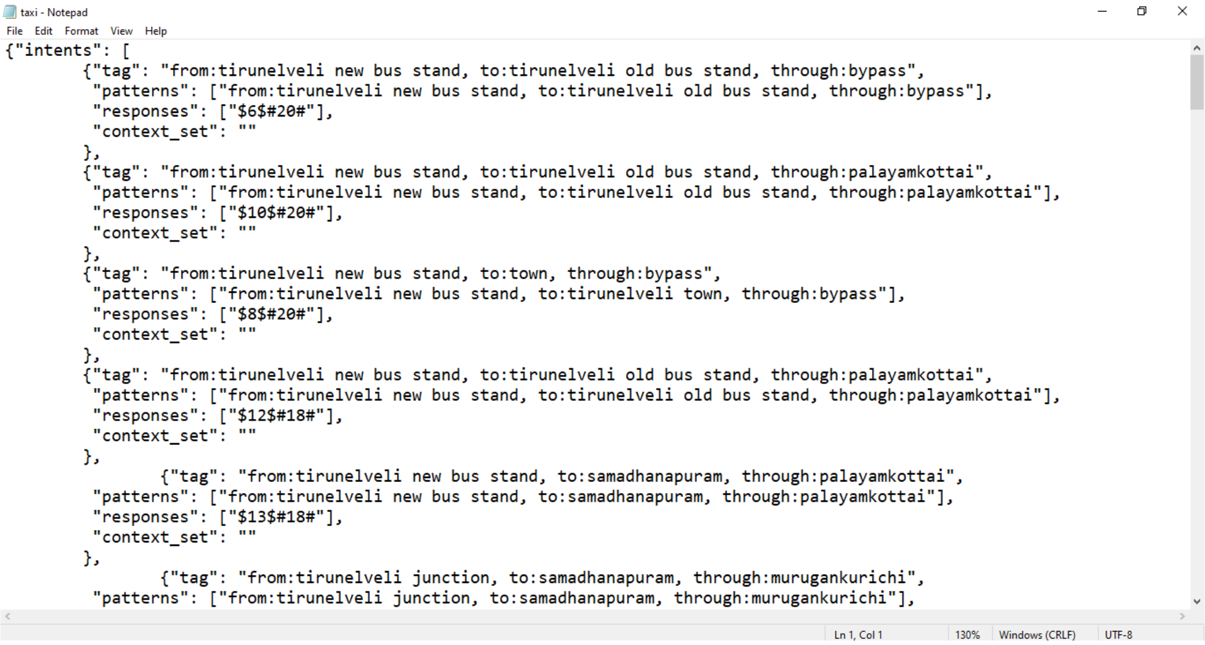
In this testing module we use the tkinter package that helps to create user interface in the application and create effective graphical user interface. This framework provides Python users with a simple way to create GUI elements using the widgets found in the Tk toolkit. Tk widgets can be used to construct buttons, menus, data fields, etc. in a Python application. Once created, these graphical elements can be associated with or interact with features, functionality, methods, data or even other widgets. To design the window of the application by providing title, size and the configuration of color of the window is given. For including label with the editable text with its width and height is given, the text input for price is also provided in the dialog window.

To get the output the submit button is given, we are providing the labels and texts in the grid form that the row and columns are specified. The main goal of the module is to check the fraud detection with the predicted price to the taxi driver charges for the passenger to check the fraud the function checktaxifraud function is used. Pickle module is used for serializing and de-serializing a Python object structure. Any object in Python can be pickled so that it can be saved on disk. What pickle does is that it "serializes" the object first before writing it to file. Pickling is a way to convert a python object (list, dict, etc.) into a character stream. The idea is that this character stream contains all the information necessary to reconstruct the object in another python script. we imported the knn pickle file that is the trained model used for prediction. The input data given by the user that will be received and changed into bag of words. These bag of words for the particular input is classified and identified from the bag of words then their responses for the input is provided as price for the trip, total distance for the travel, Taxi fraud is detected or not is provided int the window. For example it was the age of wisdom" = [1, 1, 1, 0, 1, 0, 0, 1, 1, 0]. All ordering of the words is nominally discarded and we have a consistent way of extracting features from any document in our corpus, ready for use in modeling. New documents that overlap with the vocabulary of known words, but may contain words outside of the vocabulary, can still be encoded, where only the occurrence of known words are scored and unknown words are ignored.A simple text cleaning techniques that can be used as a first step, such as: Ignoring case,Ignoring punctuation,Ignoring frequent words that don't contain much information, called stop words, like "a," "of," etc.Fixing misspelled words. Reducing words to their stem (e.g. "play" from "playing") using stemming algorithms.A more sophisticated approach is to create a vocabulary of grouped words. This both changes the scope of the vocabulary and allows the bag-of-words to capture a little bit more meaning from the document.

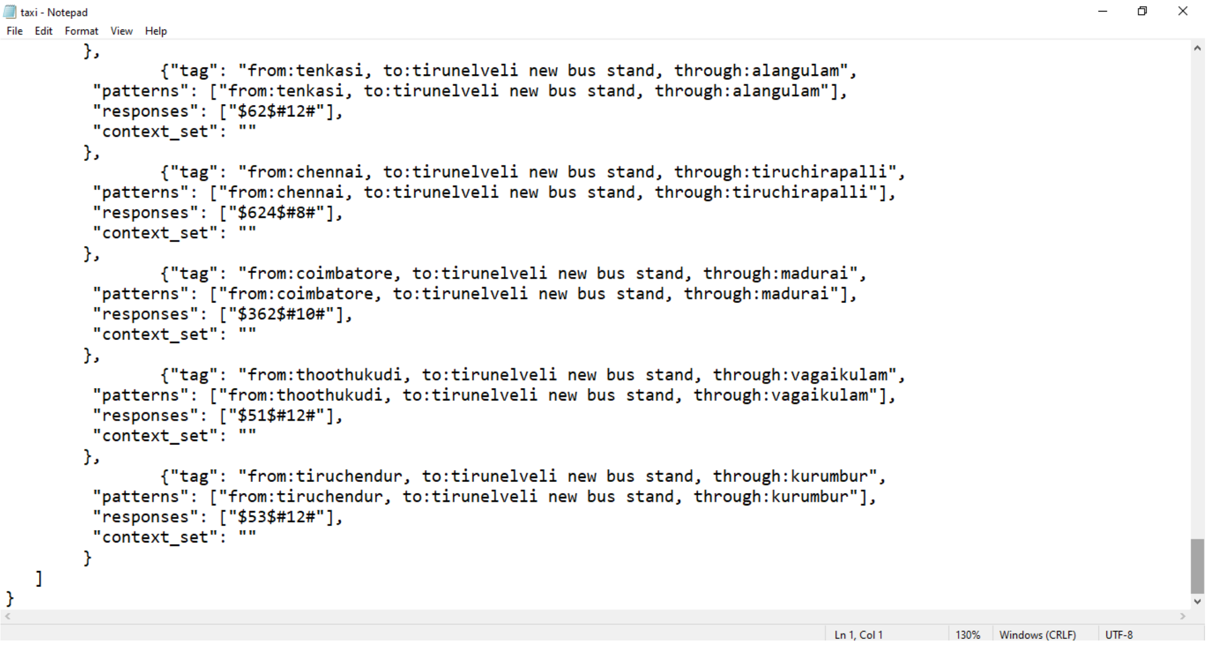
The result response searches the similar data from the dataset and provide the output. If the input is not given properly to display the message please fill all fields are given in the window. When the testing module gets executed an window will be opened the fields like from, to ,through, one way or round trip, AC or Non AC and the price asked by the taxi driver are asked in the window. When the data is entered in the window then submit button is clicked a pop up window is displayed with the details of boarding place, destination and through place is provided. Then the total distance from source and destination, price per kilometre, total amount for the trip, total amount with AC and Non AC, Total Amount for up and down. If the fraud is detected or not will be displayed at the end. In this way the testing module works in this application.

CHAPTER 4

RESULTS

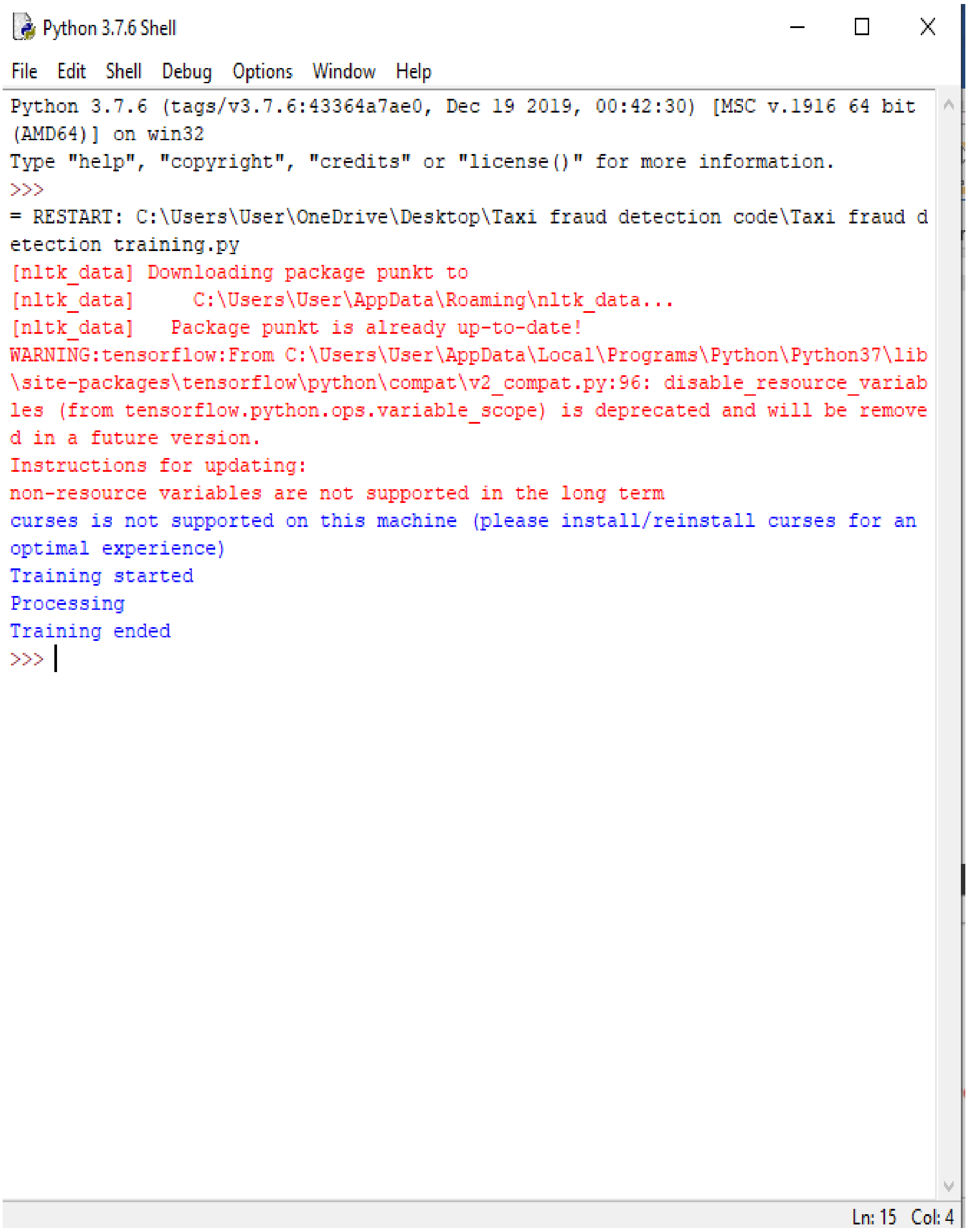


```
{"intents": [
        {"tag": "from:tirunelveli new bus stand, to:tirunelveli old bus stand, through:bypass",
         "patterns": ["from:tirunelveli new bus stand, to:tirunelveli old bus stand, through:bypass"],
         "responses": ["$6$#20#"],
         "context_set": ""
        },
        {"tag": "from:tirunelveli new bus stand, to:tirunelveli old bus stand, through:palayamkottai",
         "patterns": ["from:tirunelveli new bus stand, to:tirunelveli old bus stand, through:palayamkottai"],
         "responses": ["$10$#20#"],
         "context_set": ""
        },
        {"tag": "from:tirunelveli new bus stand, to:town, through:bypass",
         "patterns": ["from:tirunelveli new bus stand, to:tirunelveli town, through:bypass"],
         "responses": ["$8$#20#"],
         "context_set": ""
        },
        {"tag": "from:tirunelveli new bus stand, to:tirunelveli old bus stand, through:palayamkottai",
         "patterns": ["from:tirunelveli new bus stand, to:tirunelveli old bus stand, through:palayamkottai"],
         "responses": ["$12$#18#"],
         "context_set": ""
        },
                {"tag": "from:tirunelveli new bus stand, to:samadhanapuram, through:palayamkottai",
         "patterns": ["from:tirunelveli new bus stand, to:samadhanapuram, through:palayamkottai"],
         "responses": ["$13$#18#"],
         "context_set": ""
        },
                {"tag": "from:tirunelveli junction, to:samadhanapuram, through:murugankurichi",
         "patterns": ["from:tirunelveli junction, to:samadhanapuram, through:murugankurichi"],
```



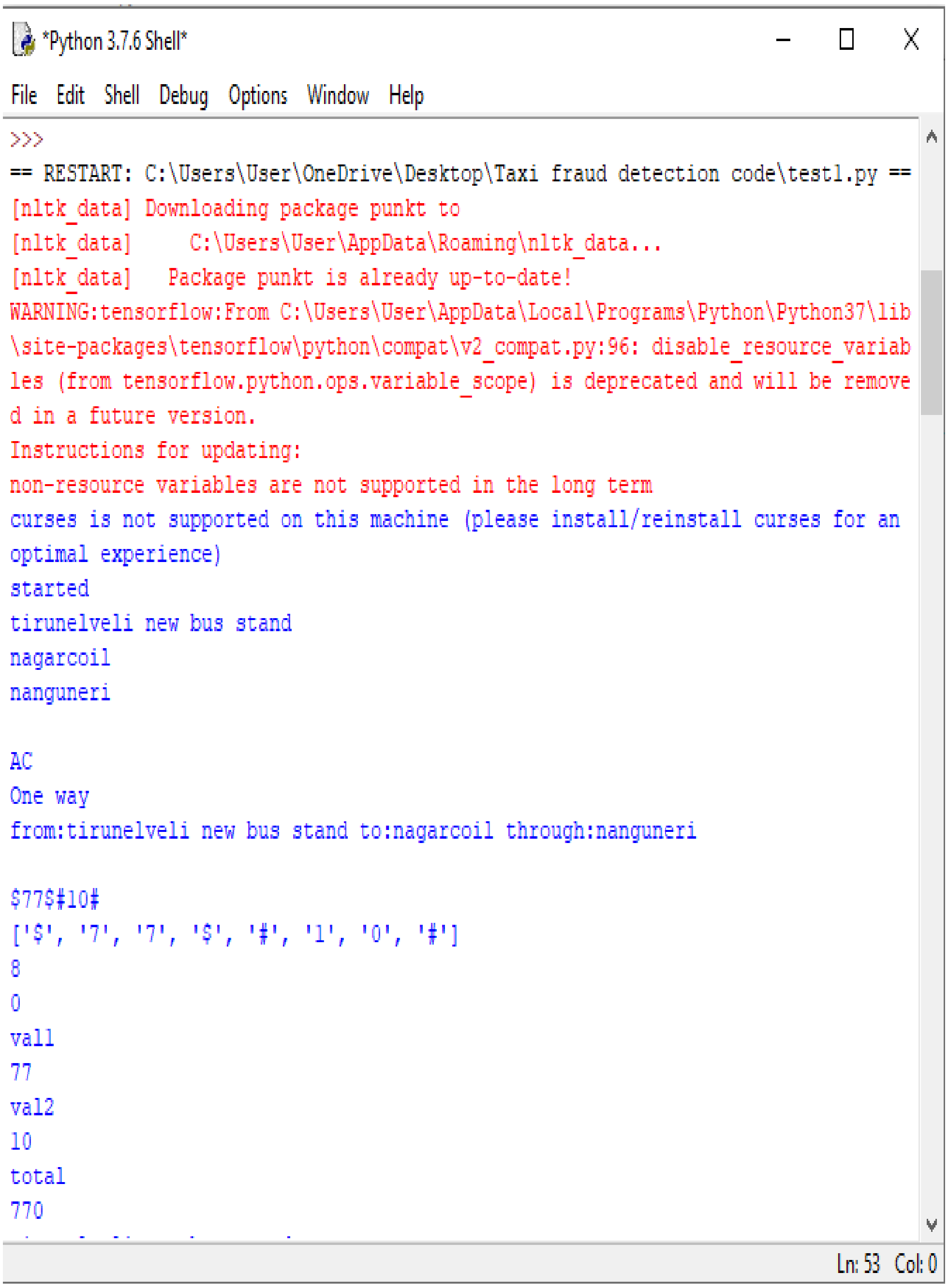
```
        },
                {"tag": "from:tenkasi, to:tirunelveli new bus stand, through:alangulam",
         "patterns": ["from:tenkasi, to:tirunelveli new bus stand, through:alangulam"],
         "responses": ["$62$#12#"],
         "context_set": ""
        },
                {"tag": "from:chennai, to:tirunelveli new bus stand, through:tiruchirapalli",
         "patterns": ["from:chennai, to:tirunelveli new bus stand, through:tiruchirapalli"],
         "responses": ["$624$#8#"],
         "context_set": ""
        },
                {"tag": "from:coimbatore, to:tirunelveli new bus stand, through:madurai",
         "patterns": ["from:coimbatore, to:tirunelveli new bus stand, through:madurai"],
         "responses": ["$362$#10#"],
         "context_set": ""
        },
                {"tag": "from:thoothukudi, to:tirunelveli new bus stand, through:vagaikulam",
         "patterns": ["from:thoothukudi, to:tirunelveli new bus stand, through:vagaikulam"],
         "responses": ["$51$#12#"],
         "context_set": ""
        },
                {"tag": "from:tiruchendur, to:tirunelveli new bus stand, through:kurumbur",
         "patterns": ["from:tiruchendur, to:tirunelveli new bus stand, through:kurumbur"],
         "responses": ["$53$#12#"],
         "context_set": ""
        }
   ]
}
```

Fig 4.1 Dataset file(taxi.jason) contains the input data values, machine readable values and responses. the tag and patterns consists of the boarding point(from),destination(to) and the route to the destination (through) data's are provided. The response stores the total kilometres and the price per kilometre.



```
Python 3.7.6 (tags/v3.7.6:43364a7ae0, Dec 19 2019, 00:42:30) [MSC v.1916 64 bit
(AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> 
= RESTART: C:\Users\User\OneDrive\Desktop\Taxi fraud detection code\Taxi fraud d
etection training.py
[nltk_data] Downloading package punkt to
[nltk_data]     C:\Users\User\AppData\Roaming\nltk_data...
[nltk_data]   Package punkt is already up-to-date!
WARNING:tensorflow:From C:\Users\User\AppData\Local\Programs\Python\Python37\lib
\site-packages\tensorflow\python\compat\v2_compat.py:96: disable_resource_variab
les (from tensorflow.python.ops.variable_scope) is deprecated and will be remove
d in a future version.
Instructions for updating:
non-resource variables are not supported in the long term
curses is not supported on this machine (please install/reinstall curses for an
optimal experience)
Training started
Processing
Training ended
>>> 
```

Fig 4.2 Training window that displays the training process starting and ending. The whole process starts with importing the dataset and ends in the training of the dataset by converting the text data to numerical values for the prediction by the KNN algorithm.

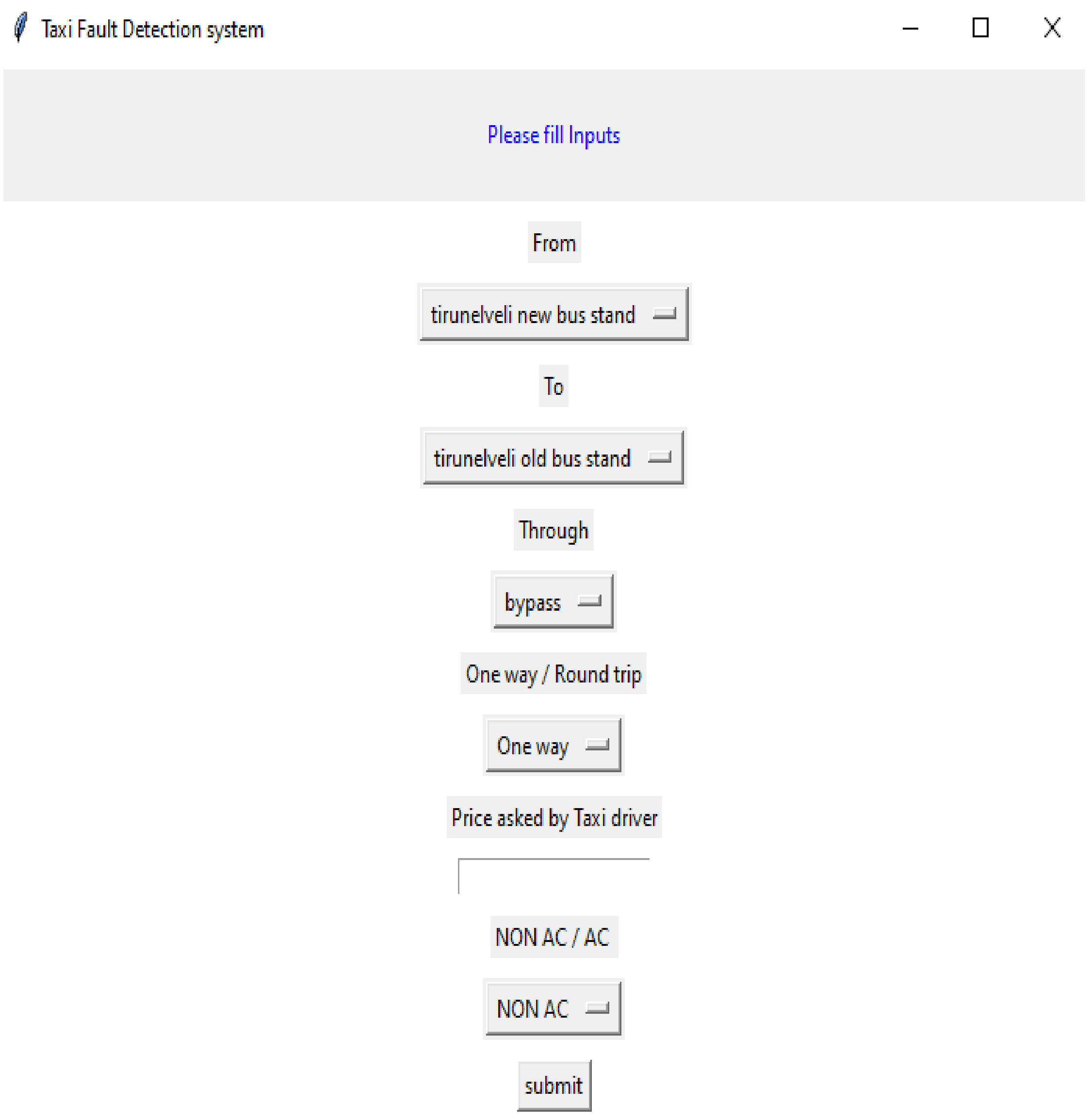


```
>>>
== RESTART: C:\Users\User\OneDrive\Desktop\Taxi fraud detection code\test1.py ==
[nltk_data] Downloading package punkt to
[nltk_data]     C:\Users\User\AppData\Roaming\nltk_data...
[nltk_data]   Package punkt is already up-to-date!
WARNING:tensorflow:From C:\Users\User\AppData\Local\Programs\Python\Python37\lib
\site-packages\tensorflow\python\compat\v2_compat.py:96: disable_resource_variab
les (from tensorflow.python.ops.variable_scope) is deprecated and will be remove
d in a future version.
Instructions for updating:
non-resource variables are not supported in the long term
curses is not supported on this machine (please install/reinstall curses for an
optimal experience)
started
tirunelveli new bus stand
nagarcoil
nanguneri

AC
One way
from:tirunelveli new bus stand to:nagarcoil through:nanguneri

$77$#10#
['$', '7', '7', '$', '#', '1', '0', '#']
8
0
val1
77
val2
10
total
770
```

Fig 4.3 Testing window - the execution of the input data is being calculated for the trip that is the total distance ,price also shown



Taxi Fault Detection system

Please fill Inputs

From

tirunelveli new bus stand

To

tirunelveli old bus stand

Through

bypass

One way / Round trip

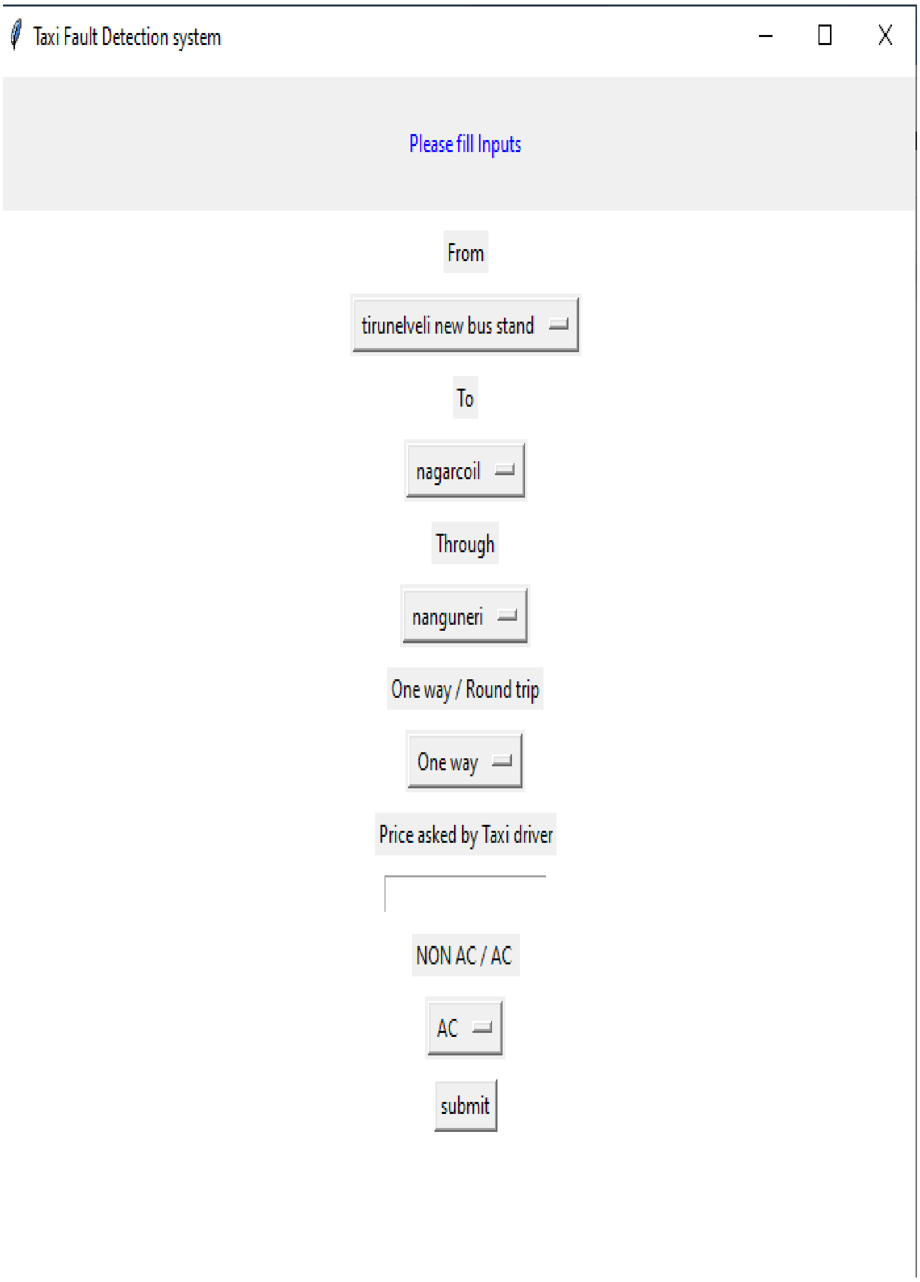
One way

Price asked by Taxi driver

NON AC / AC

NON AC

Fig 4.4 User interface window to get the input values, This window that initial displays with the predefined places before the passenger gives the input. It has fields like from, to ,through, one way or round trip, AC or Non AC. if the taxi driver asks charges for the ride then to enter that value in the Price asked by the taxi driver to detect Fraud.



Taxi Fault Detection system

Please fill Inputs

From

tirunelveli new bus stand

To

nagarcoil

Through

nanguneri

One way / Round trip

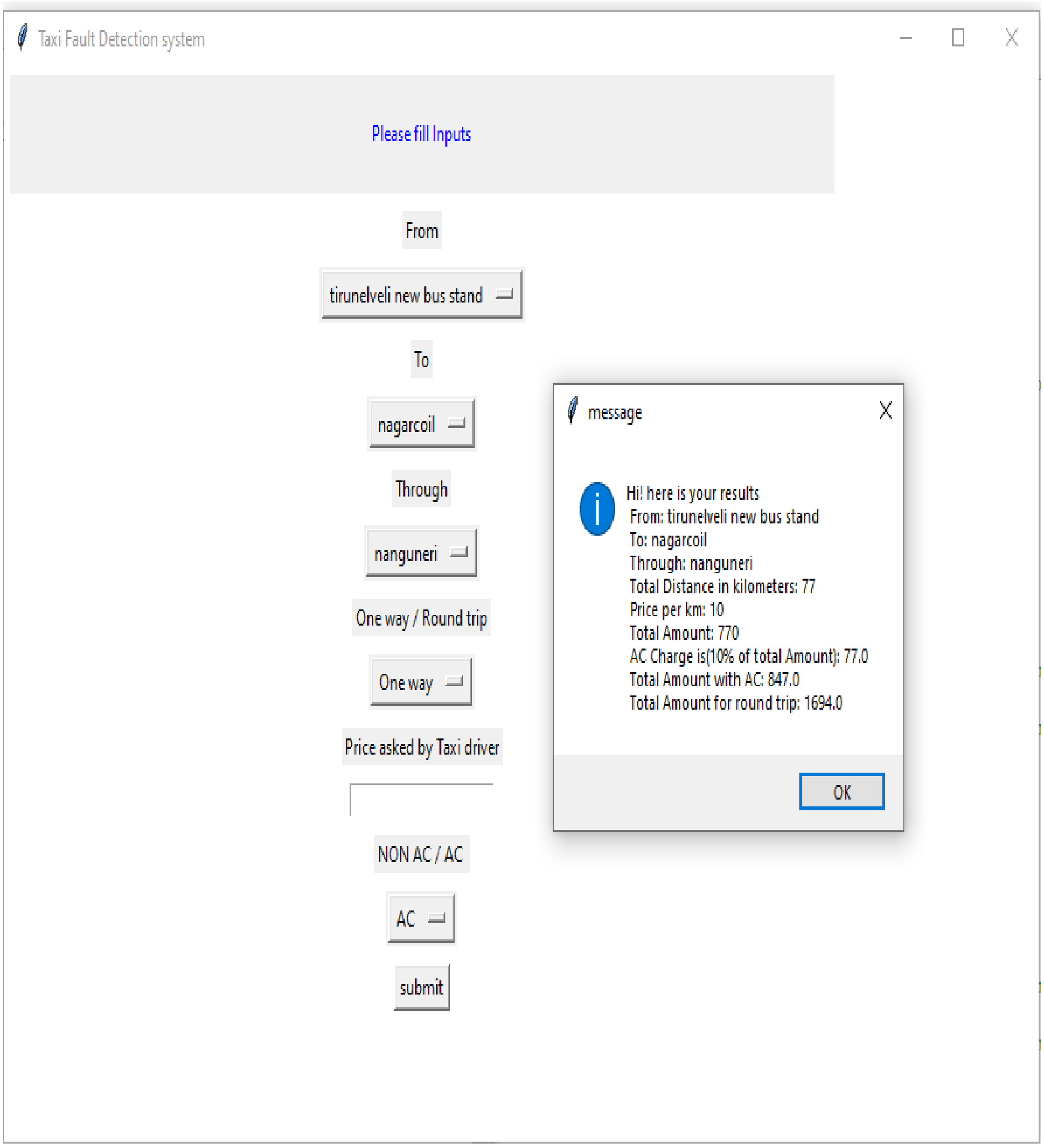
One way

Price asked by Taxi driver

NON AC / AC

AC

Fig 4.5 User interface window with the input values enter. Here the passenger opted the boarding place as the Tirunelveli New Bus stand to reach Nagercoil through the place naguneri for oneway trip with Ac .



Please fill Inputs

From
tirunelveli new bus stand

To
nagarcoil

Through
nanguneri

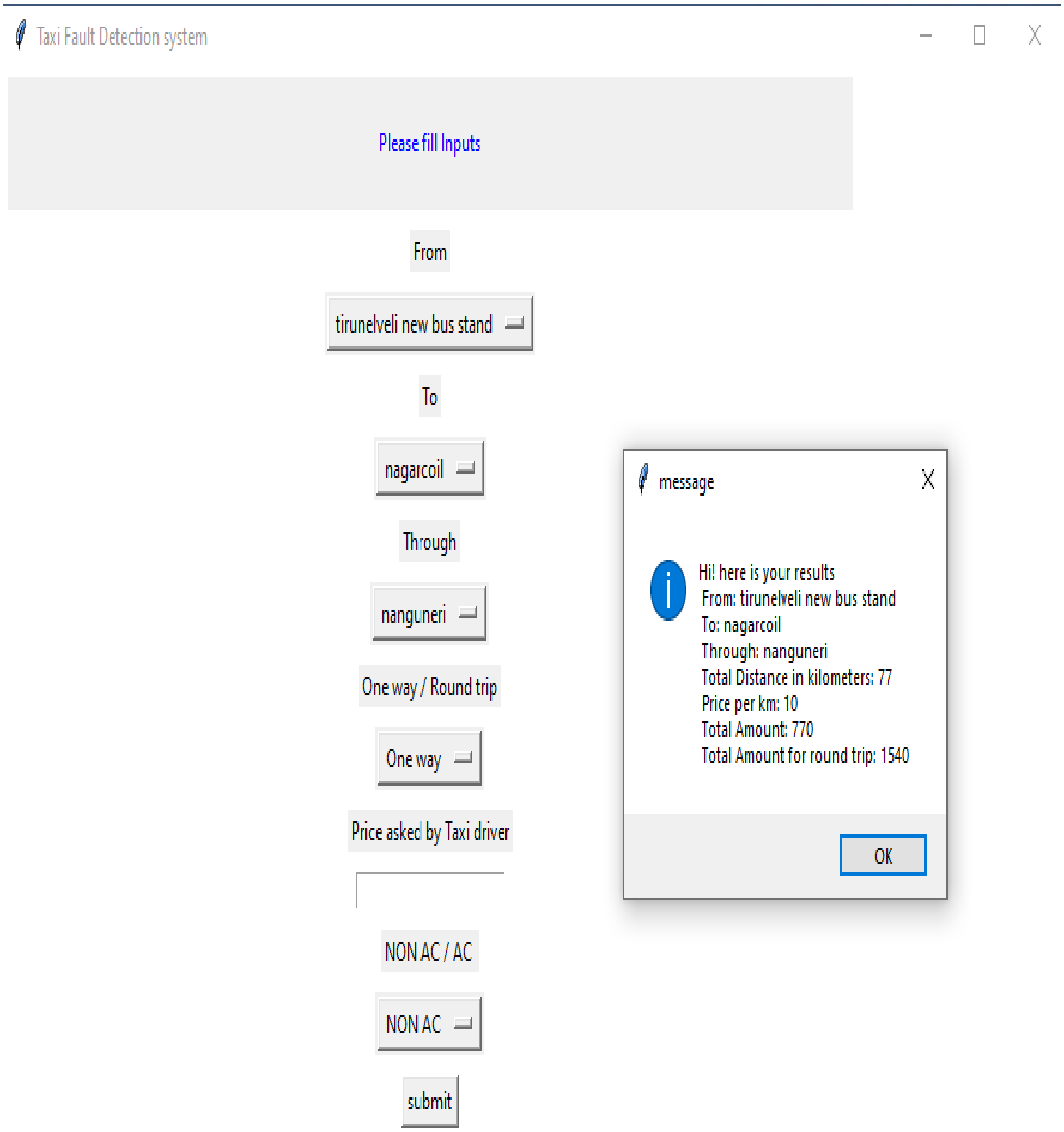
One way / Round trip
One way

Price asked by Taxi driver

NON AC / AC
AC

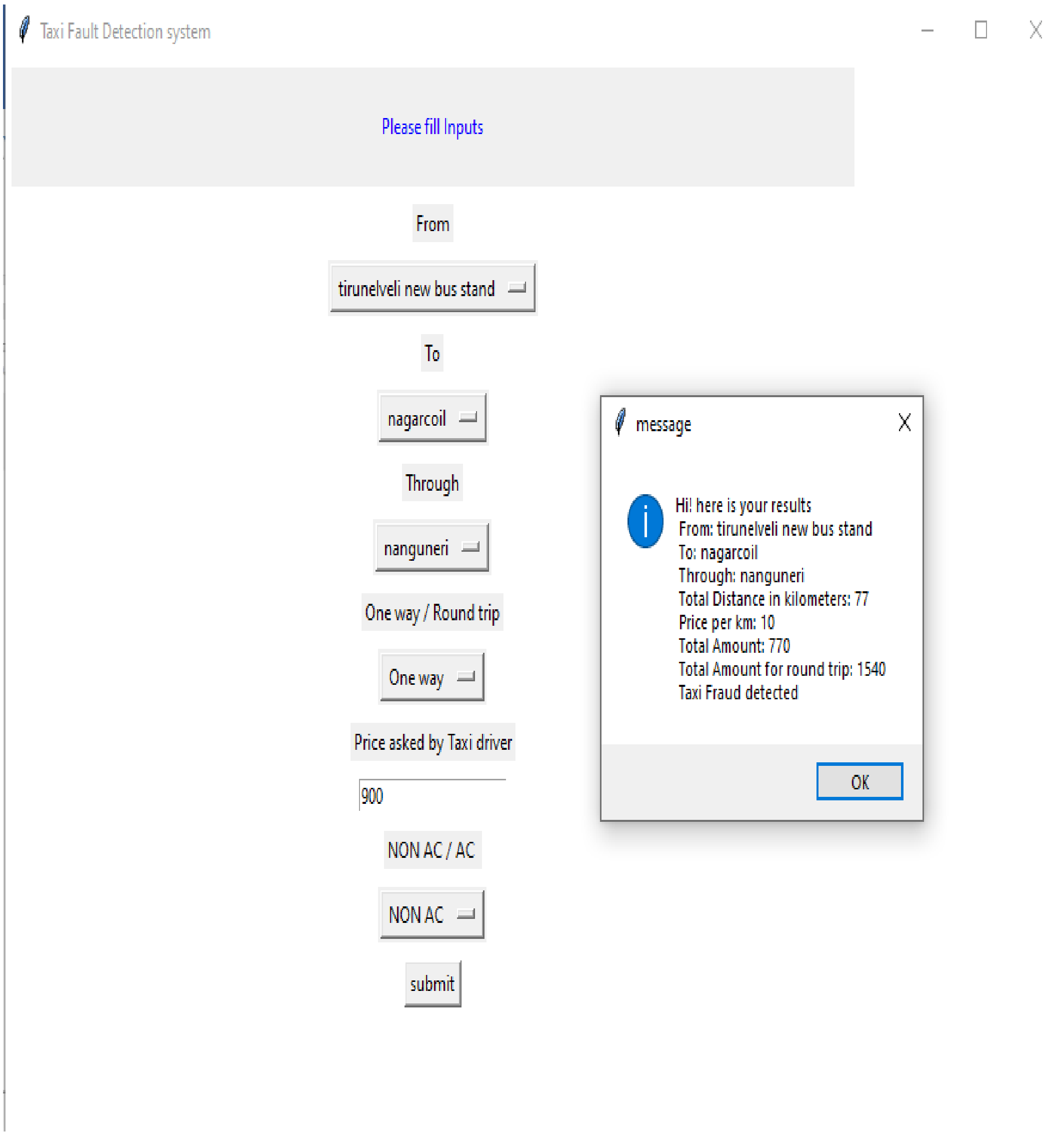
```
Hi! here is your results
 From: tirunelveli new bus stand
 To: nagarcoil
 Through: nanguneri
 Total Distance in kilometers: 77
 Price per km: 10
 Total Amount: 770
 AC Charge is(10% of total Amount): 77.0
 Total Amount with AC: 847.0
 Total Amount for round trip: 1694.0
```

Fig 4.6 Taxi Fraud Detection window with price prediction for Ac trip. After the submission of input is provided by the user. A small dialog box displayed with the details of the trip. Provides the total distance of the trip, Price per km, AC Charge 10% of the total amount, Total Amount with AC.



```
Hi! here is your results
From: tirunelveli new bus stand
To: nagarcoil
Through: nanguneri
Total Distance in kilometers: 77
Price per km: 10
Total Amount: 770
Total Amount for round trip: 1540
```

Fig 4.7 window with price prediction for Non AC. For this one way trip with Non AC the amount is calculated and is displayed.



From: tirunelveli new bus stand

To: nagarcoil

Through: nanguneri

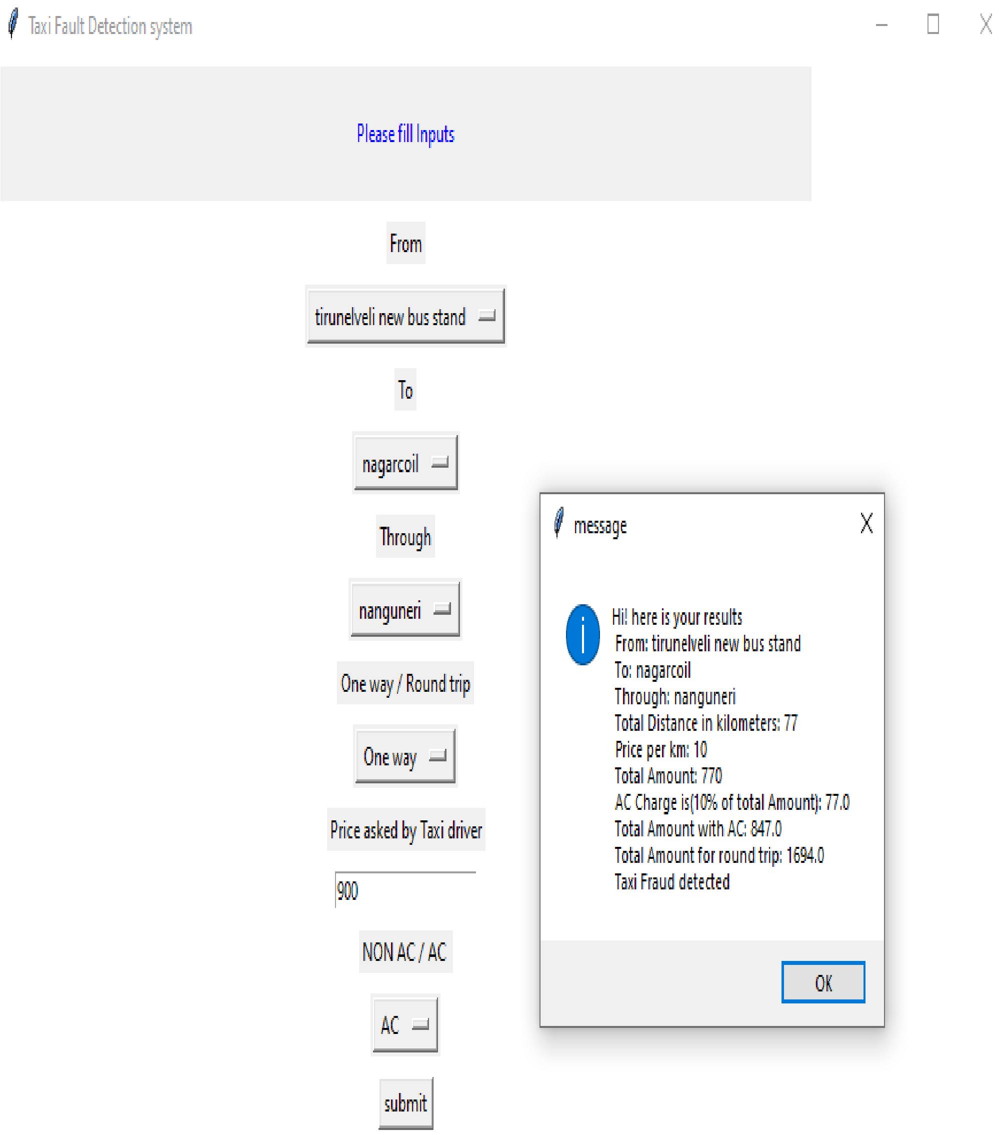
One way / Round trip: One way

Price asked by Taxi driver: 900

NON AC / AC: NON AC

```
Hi! here is your results
From: tirunelveli new bus stand
To: nagarcoil
Through: nanguneri
Total Distance in kilometers: 77
Price per km: 10
Total Amount: 770
Total Amount for round trip: 1540
Taxi Fraud detected
```

Fig 4.8 Fraud detection for the price given by the driver with Non AC. The price asked by the taxi driver is given, the machine checks whether charge asked by the driver is more than the predicted amount or not. If it is high then fraud is identified. For Non AC the asked amount is higher so the taxi fraud is detected.



From: tirunelveli new bus stand

To: nagarcoil

Through: nanguneri

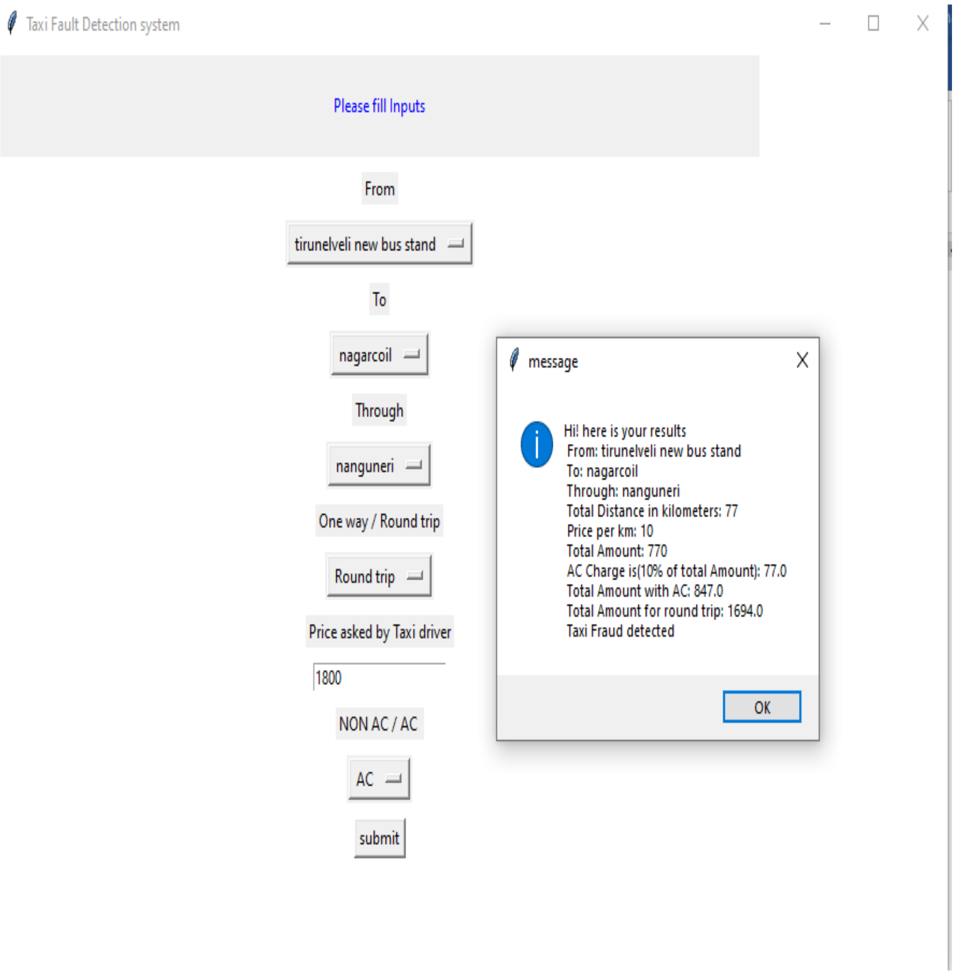
One way / Round trip: One way

Price asked by Taxi driver: 900

NON AC / AC: AC

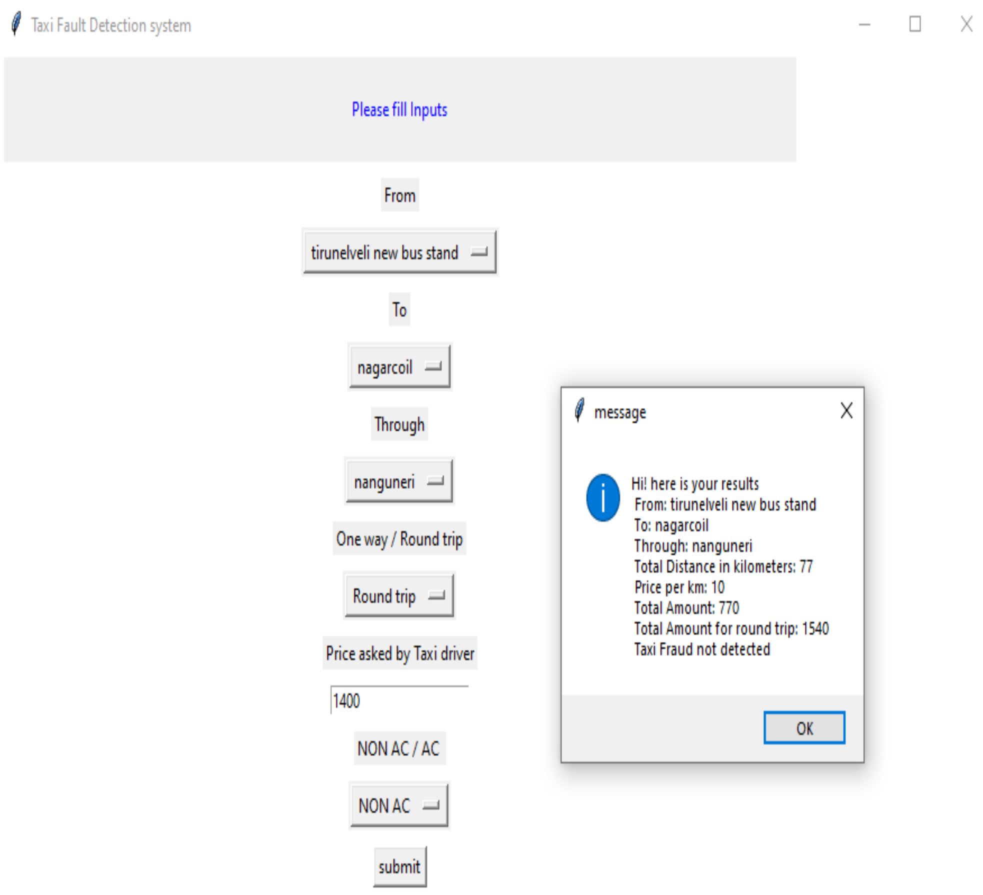
```
Hi! here is your results
From: tirunelveli new bus stand
To: nagarcoil
Through: nanguneri
Total Distance in kilometers: 77
Price per km: 10
Total Amount: 770
AC Charge is(10% of total Amount): 77.0
Total Amount with AC: 847.0
Total Amount for round trip: 1694.0
Taxi Fraud detected
```

Fig 4.9 fraud detection for the price given by the driver with AC. The price asked for the trip with AC is higher that the predicted amount .So the taxi fraud trip is detected



Hi! here is your results
From: tirunelveli new bus stand
To: nagarcoil
Through: nanguneri
Total Distance in kilometers: 77
Price per km: 10
Total Amount: 770
AC Charge is(10% of total Amount): 77.0
Total Amount with AC: 847.0
Total Amount for round trip: 1694.0
Taxi Fraud detected

Fig 4.10 Fraud detection for price given by the driver with AC for Round trip. The price asked for the round trip with AC is higher than the predicted amount so the taxi fraud detected.



From
tirunelveli new bus stand
To
nagarcoil
Through
nanguneri
One way / Round trip
Round trip
Price asked by Taxi driver
1400
NON AC / AC
NON AC

Hi! here is your results
From: tirunelveli new bus stand
To: nagarcoil
Through: nanguneri
Total Distance in kilometers: 77
Price per km: 10
Total Amount: 770
Total Amount for round trip: 1540
Taxi Fraud not detected

Fig 4.11 Fraud detection for Round trip with Non AC. The price asked by the taxi driver for the round trip is less than the predicted output. So taxi fraud not detected.

CONCLUSION

In this project we considered metered and unmetered taxi trip in real-world data, with the distance and price. To find the charges made by the taxi driver is acceptable or not. We proposed the novel system Fiddle Tour: Fraudulent Taxi Trip Detection using KNN Machine Learning Algorithm which predicts the Taxi fraud with the help of distance and price for the path travelled. The Dataset is trained by the KNN Algorithm to find The response for the specific Travel path given by the passenger. To get the response according to the requirements of the passenger like one way trip, Round trip, AC, Non AC and the price for the trip also provided. By using this Algorithm the efficiency of prediction of fraud trip is high and it executes faster. In this project the fraud trip prediction is only done we have not visualized the path of the trip, which will be studied and implemented in our future works.

REFERENCE

[1] Ye Ding, Member, IEEE, Wenyi Zhang, Member, IEEE, Xibo Zhou, Member, IEEE, Qing Liao , Member, IEEE, Qiong Luo, Member, IEEE, Lionel M. Ni, Fellow, IEE "FraudTrip: Taxi Fraudulent Trip Detection from Corresponding Trajectories" 2019 IEEE Internet of Things Journal

[2] Asma Belhadi, Youcef Djenouri , Gautam Srivastava , Senior Member, IEEE, Djamel Djenouri, Alberto Cano , Senior Member, IEEE, and Jerry Chun-Wei Lin , Senior Member," A Two-Phase Anomaly Detection Model for Secure Intelligent Transportation Ride-Hailing Trajectories" Volume22, 2021 IEEE Transactions on intelligent transportation systems.

[3] Chao Chen, Daqing Zhang, Member, IEEE, Pablo Samuel Castro, Nan Li, Lin Sun, Shijian Li, and Zonghui Wang "iBOAT: Isolation-Based Online Anomalous Trajectory Detection"Volume 14, 2019 IEEE transactions on intelligent transportation systems.

[4] Wei Tu, Member, IEEE, Mai Ke, Yatao Zhang, Yang Xu, Jincai Huang, Min Deng, Long Chen, Senior Member, IEEE, Qingquan Li "Real-time Route Recommendations for E-Taxies Leveraging GPS Trajectories" Volume 17, 2021

IEEE Transactions on Industrial Informatics.

[5] Yongxuan Lai , Zheng Lv , Kuan-Ching Li, and Minghong Liao

"Urban Traffic Coulomb's Law: A New Approach for Taxi Route Recommendation" Volume: 20, Aug. 2019 IEEE Transactions on intelligent transportation system.

[6] Huigui Rong1 , (Member, IEEE), Zepeng Wang1 , Hui Zheng2 , Chunhua Hu2 , (Member, IEEE) Li Peng1 , Zhaoyang Ai3 , Arun Kumar Sangaiah "Mining efficient taxi operation strategies from large scale geo-location data" Volume 5, 2017 IEEE Access

[7]Jinglin Li, Member, IEEE, Dawei Fu, Quan Yuan, Haohan Zhang, Kaihui Chen, Shu Yang, and Fangchun Yang, Senior Member, IEEE " A Traffic Prediction Enabled Double Rewarded Value Iteration Network for Route Planning" Volume 7, 2019 IEEE Transactions on Vehicular Technology

[8] Li Li , Fellow, IEEE, Shuofeng Wang, and Fei-Yue Wang, Fellow, IEEE

"An Analysis of Taxi Driver's Route Choice Behavior Using the Trace Records"

VOL. 5, NO. 2, JUNE 2018 IEEE Transactions On Computational Social Systems.

[9] Yan Lyu, Victor C. S. Lee, Member, IEEE, Joseph K. Y. Ng, Senior Member, IEEE, Brian Y. Lim, Kai Liu, Member, IEEE, Chao Chen, Member, IEEE, Flexi-Sharing: A "Flexible and Personalized Taxi-Sharing System" Volume: 68, Oct. 2019,IEEE Transactions on Vehicular Technology

[10] Xiangjie Kong , Senior Member, IEEE, Menglin Li, Tao Tang , Kaiqi Tian, Luis Moreira-Matias , Member, IEEE, and Feng Xia , Senior Member, IEEE

"Shared Subway Shuttle Bus Route Planning Based on Transport Data Analytics" Volume 15 2018,IEEE Transactions on automation science and engineering

[11]X. Zhou, Y. Ding, F. Peng, Q. Luo, and L. M. Ni, "Detecting unmetered taxi rides from trajectory data," in 2017 IEEE International Conference on Big Data (Big Data). IEEE, 2017, pp. 530–535.

[12] 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.

[13] 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.

[14] J. Yuan, Y. Zheng, C. Zhang, W. Xie, X. Xie, G. Sun, and Y. Huang, "Tdrive: 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.

[15] 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.

[16] J. W. Powell, Y. Huang, F. Bastani, and M. Ji, "Towards reducing taxicab cruising time using spatio-temporal profitability maps." in SSTD.Springer, 2011, pp. 242–260.

[17] K. Yamamoto, K. Uesugi, and T. Watanabe, "Adaptive routing of cruising taxis by mutual exchange of pathways," in International Conferenceon Knowledge-Based and Intelligent Information and EngineeringSystems. Springer, 2008, pp. 559–566.

[18] B. Li, D. Zhang, L. Sun, C. Chen, S. Li, G. Qi, and Q. Yang,"Hunting or waiting? discovering passenger-finding strategies froma large-scale real-world taxi dataset," in Pervasive Computing and Communications Workshops (PERCOM Workshops), 2011 IEEEnternational Conference on. IEEE, 2011, pp. 63–68.

[19] 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.

[20] J. Li, G. Luo, N. Cheng, Q. Yuan, Z. Wu, S. Gao, and Z. Liu, "An end-to-end load balancer based on deep learning for vehicular network traffic control," IEEE Internet Things J., 2018, DOI: 10.1109/JIOT.2018.2866435.

[21] 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. [2] C. G. Prato, "Route choice modeling: Past, present and future research directions," J. Choice Model., vol. 2, no. 1, pp. 65–100, 2009

[22] 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

[23] S. Liu, L. M. Ni, and R. Krishnan, "Fraud detection from taxis' driving behaviors," IEEE Transactions on Vehicular Technology, vol. 63, no. 1, pp. 464–472, 2013.

[24] 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.

[25] S. Zhang and Z. Wang, "Inferring passenger denial behavior of taxi drivers from large-scale taxi traces," PloS one, vol. 11, no. 11, 2016.

[26]J.-G. Lee, J. Han, and X. Li, "Trajectory outlier detection: A partition and detect framework," in Data Engineering, 2008. ICDE 2008. IEEE 24th International Conference on. IEEE, 2008, pp. 140–149. [8] Y. Bu, L. Chen, A. W.-C. Fu, and D. Liu, "Efficient anomaly monitoring over moving object trajectory

[27]streams," in Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2009, pp. 159–168.

[28]J. Yang, J. Dong, Z. Lin, and L. Hu, "Predicting market potential and environmental benefits of deploying electric taxis in nanjing, china," Transport. Res. Part D-Transport. Environ., vol. 49, pp. 68–81, 2016.

[29] L. Zhu, F. R. Yu, Y. Wang, B. Ning, and T. Tang, "Big data analytics in intelligent transportation systems: A survey," IEEE Trans. Intell. Transp. Syst., vol. 20, no. 1, pp. 383–398, Jan. 2019.

[30] I. Kalamaras et al., "An interactive visual analytics platform for smart intelligent transportation systems management," IEEE Trans. Intell. Transp. Syst., vol. 19, no. 2, pp. 487–496, Feb. 2018.

[31] B. Qu, W. Yang, G. Cui, and X. Wang, "Profitable taxi travel route recommendation based on big taxi trajectory data," IEEE Trans. Intell. Transp. Syst., 2019.