Accident Prediction System Using Machine Learning

Bhavani Shankar[1], Charan H V[2], Chitrashwa R[3], Hemal Jayachander[4]

[1,2,3,4] *Bachelor of Engineering, Computer Science & Engineering, Bangalore Institute of technology, Bangalore, Karnataka, India.*

Under the guidance of ***Sushma H R****, Asst. Professor, Department of Computer Science & Engineering, Bangalore Institute of technology, Bangalore, Karnataka, India.*

---***---

Abstract - *The traffic has been transformed into the deranged issue in modern life due to increasing number of vehicles. Which leads to accidents. Despite all that has been done to increase Road Safety in India so far, there are always regions that fall victim to the vulnerabilities that present in every corner. The issue leads to the need for an effective analysis so as to reduce the alarming figures by a significant amount. The objective of this paper is to have machine learning algorithm to come to aid to create a model that not only smooths out the heterogeneity of the data by grouping similar objects together to find the accident prone areas in the city with respect to different accident-factors but also helps determine the association between these factors and casualties. This situation has discovered road accidents problem, affecting factors and remedies to be taken to prevent it.*

Key Words: **Artificial Intelligence, Data Mining, Machine Learning, K-Means, Centroid, Cluster, Euclidean Distance, Tkinter, Rule Mining.**

1.INTRODUCTION

Road accidents are one of the most frequent cause of damage in modern world. It's one of the most significant causes of the casualties. The causes for this are the extremely dense road traffic and the relatively great freedom of movement given to drivers. Accidents that involve heavy goods vehicles like Lorries, trucks and even the commercial vehicles with the public transportation like buses are one of the most fatal kinds of accidents that occur, claiming the lives of innocent people[1]. The other factors that are attributed to leading to such a mishap can range anywhere from vehicular defect and weather conditions to transportation conditions.

Highways are always attracted for these accidents with injuries and deaths. Various weather conditions like rain, fog etc., play a role in creating the risk of accidents. Having a proper estimation of accidents and knowing the hotspot of accidents and its factors will help to reduce them. Providing timely emergency support even when the casualties have occurred is needed, and to do that a keen study on accidents is required. In spite of having set regulations and the highway codes, negligence of people towards the speed of the vehicle, the vehicle condition and their own negligence of not wearing helmets has caused a lot of accidents. These accidents wouldn't have turned fatal, and claimed innocent lives if people had governed by the rules.

Machine learning which is a sub-branch of artificial intelligence supplies learning of computer taking advantage of data warehouses. Assumption or classification abilities of computer systems have advanced in the event of machine learning. Utilization of machine learning is a extensive and functional method for taking veritable decisions by using experience. Machine learning is able to accomplish extracting information from data and use statistical method. In this paper, we study the metropolitical city Bangalore contributing causes, Road structure, Environment factors and Road conditions to draw efficient conclusions in order to facilitate road safety in the country. We are focused on taking the aid of clustering to group similar objects off this dataset in order to group regions on the basis of vulnerability [6]. The clusters so formed are labelled to be further classified using K-Means to give the accidental zones in the city [3].

1.1 Existing Model

The traffic has been transformed into the grueling structure in points of designing and managing by the reason of increasing number of vehicles. Large regulated data collections have increased by the reasons of the technological improvements and data storage with low cost. Arising the need of elevation to information from this large calibrated data obtained the corner stone of the data mining. In this study, the most compatible machine learning classification techniques for road accidents estimation by data mining has been intended to study and compare.

1.2 Objectives

The main objective of our road accident prediction system:

- Analyse the previously occurred accidents in the locality which will help us to determine the most accident-prone area and help us to set up the immediate required help for them.
- To make predictions supported constraints like weather, pollution, road structure, etc.

1.3 Problem Statement

There are several problems with current practices for prevention of the accidents occurred within the localities. The database we'll use is available officially by many institutes and government websites which is known n popular for data. The data collected will be analysed, implemented and grouped together based on different constraints considering important using the best suited algorithm. This estimation will be helpful to examine and recognize the flaws and the reasons of the accidents. It will also be helpful while making roads and bridges as a reference to avoid the same problems faced before and to build it in better way. The predictions made will be very much useful for management to overcome such problems. To develop a model to identify accident prone zone areas and to classify the them as high risk and low risk areas.

2. METHODOLOGY

K-MEANS Clustering is used to group similar object off of the heterogeneous data. As per this algorithm, an object can be allotted to only one cluster. Euclidean distance is the measure used to define the centroid of a cluster. K is the number of clusters and is usually given a small integer value (1, 2, 3…). K points are then chosen randomly-preferably the initial ones which represent the centroids of k clusters without any members and are placed to the cluster with the centroid nearest to it [2]. The less variation we have within clusters, the more homogeneous or similar the data points are within the same cluster.

The way kmeans algorithm works in following steps:

1. Specify number of clusters *K*.
2. Initialize centroids by first rearranging the dataset and then randomly selecting *K* data points for the centroids without replacement.
3. Keep iterating until there is no difference to the centroids. i.e. assignment of next data points to clusters isn't changing.

- Compute the sum of the squared distance between all data points and all centroids.
- Assign each data point to the nearest cluster(centroid).
- Compute the centroid for the clusters by taking average of the all data points to each clusters.

The approach kmeans travel to solve the problem is called **Expectation-Maximization.** The E-step is assigning the new data points to the closest cluster. The M-step is computing the centroid of each cluster after adding new data.

Below is a detailed steps of how we can solve it mathematically.

The objective function is:

$$J = \sum_{i=1}^{m} \sum_{k=1}^{K} w_{ik} \|x^i - \mu_k\|^2 \qquad (1)$$

where wik=1 for a data point xi if it belongs to cluster *k*; otherwise, wik=0. Also, μk is that the centroid of xi's cluster. It's a minimization problem of two parts. We first minimize J w.r.t. wik and treat μk fixed. Then we minimize J w.r.t. μk and treat wik fixed. Technically speaking, we differentiate J w.r.t. wik first and update cluster assignments (*E-step*). Then we differentiate J w.r.t. μk and recompute the centroids again after the cluster assignments from previous step (*M-step*). Therefore, E-step is:

$$\frac{\partial J}{\partial w_{ik}} = \sum_{i=1}^{m} \sum_{k=1}^{K} \|x^i - \mu_k\|^2$$

$$\Rightarrow w_{ik} = \begin{cases} 1 & \text{if } k = argmin_j \|x^i - \mu_j\|^2 \\ 0 & \text{otherwise.} \end{cases} \qquad (2)$$

To rephrase it, assign the data point xi to the closest cluster calculated by its sum of squared distance from cluster's centroid.

And M-step is:

$$\frac{\partial J}{\partial \mu_k} = 2\sum_{i=1}^{m} w_{ik}(x^i - \mu_k) = 0$$

$$\Rightarrow \mu_k = \frac{\sum_{i=1}^{m} w_{ik} x^i}{\sum_{i=1}^{m} w_{ik}} \qquad (3)$$

Which translates to recomputing the centroid of each cluster to reflect the new assignments added.

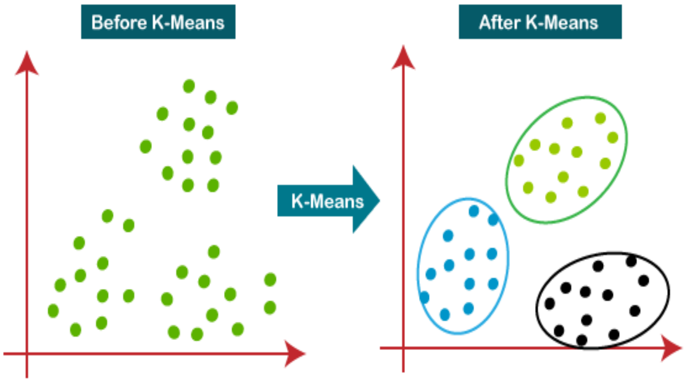


Fig 2.1.1 K-Means

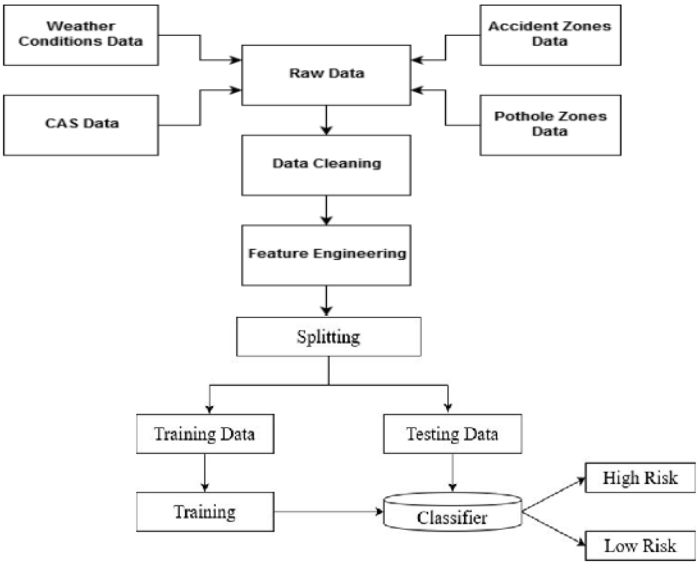


Fig 2.1.2 System architecture

The datasets obtained will undergo preprocessing. We divide the full dataset into two parts that can be either 70-30 or 80-20. The larger portion of the data sets is for the processing. The algorithm is implemented on that part of data. Which assist the algorithm to learn on its own and make prediction for the new upcoming data or the unknown data.

2.1 Module description

A module description gives definite data about the module and its upheld parts, which is open in various habits. The modules in this technique are:

- **Data Set Selection**

Data is that the most import part when you work on prediction systems. It plays a really vital role your whole project i.e., your system depends thereon data. So selection of knowledge is the first and the critical step which should be performed properly, For our project we got the info from the government website. These datasets were available for all. There are other plenty of websites who provide such data. The dataset we elect was selected based on the various factors and constraints we were going to take under the consideration for our prediction system.

- **Data Cleaning and Data Transformation**

After we've selected the dataset. the subsequent step is to clean the data and transform it into the desired format as it is possible the dataset we use may be of different format. it's also possible that we might use multiple datasets from different sources which can be in different file formats. So to use them we'd like to convert them into the format we want to or the type that type prediction system supports. the rationale behind this step is that it is possible that the data set contains the constraints which are not needed by the prediction system and including them makes the system unpredictable to learn and may extend the processing time due to noisy data. one more reason behind data cleaning is the dataset may contain null value and garbage values too. therefore the solution to this issue is when the data is transformed the garbage values are removed and null values are filled. There are many different methods to perform that.

- **Data Processing and Algorithm Implementation**

After the data sets are cleaned and transformed it's ready to process further. After the data sets has been cleaned and we have taken the required constraints. We divide the full dataset int o the two parts that can be either 70-30 or 80-20. The larger portion of data is for the processing or learning. The algorithm is applied on large part of data. Which helps the algorithm to find out on its own and make prediction for the future data or the unknown data. The algorithm is executed during which we take only the required constraints from the cleaned data. The output of the algorithm is in 'yes' and 'no' which converted to HIGH and LOW respectively . It gives the error rate as well as success rate.

- **Output and User Side Experience**

After the prediction system is prepared to use. The user just has got to login first. There is a new page with different options they need to select. They are like the predict, graphs, rules, and new data. The new data entry is used to collect new accident data from user. Once the user go to predict and clicks on "train" the algorithm is triggered and the data sets are passed to the prediction system. The user is given how accident prone the road can be in HIGH or LOW.

2.2 System Flow Chart

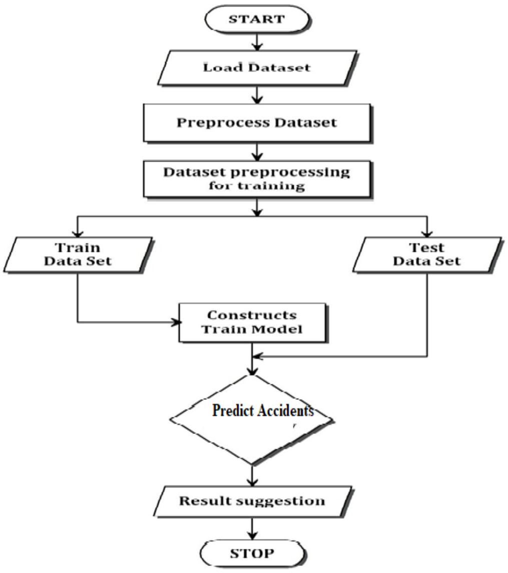


Fig 2.2 System Flow Chart

We collect the datasets and given to process in our system. We preprocess the collected data and split the data and construct the model. We train the model with larger split of data set and predict the output for new data.

2.3 Use case diagram

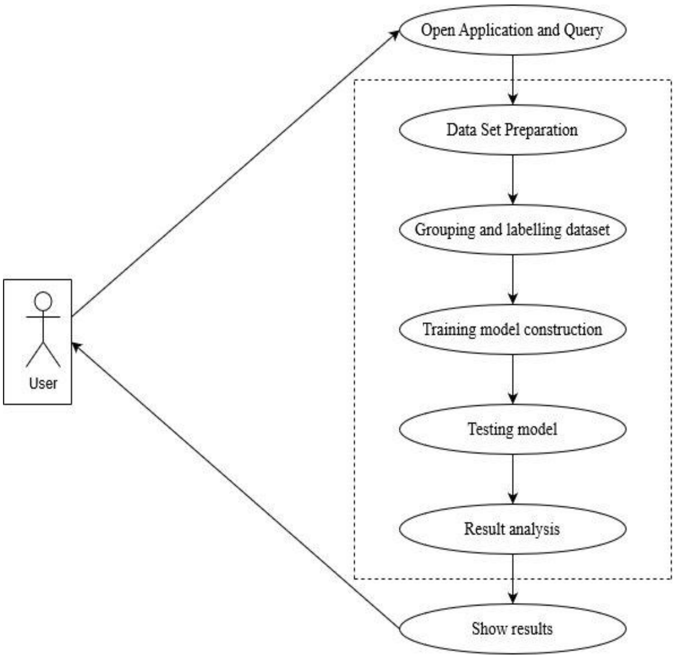

Open Application and Query
Data Set Preparation
Grouping and labelling dataset
Training model construction
Testing model
Result analysis
Show results
User

Fig 2.2 Use case diagram

As per user perspective the diagram depicts the steps involed. User first open our application. He was given to choose the option predict the accident prone zone. Once the user clicks, the data preprocessing and cleaning will takes place but hidden to the user. At first time the user is asked to click on train the model. The testing is also considered while doing the model construction. Once the model is constructed it's ready to use for new data. the user is asked to select the area from the drop down and the model predicts whether the area is prone zone or not. The result is collected and can be used for further improvements. Once the purpose is fullfilled the application can be closed.

3. Software and Languages used

3.1 Jupyter

Jupyter exists to develop open-source software. it's used for open-standards, and services for interactive computing across dozens of programming languages. it's an opensource web application that allows you to create and share documents and code live. Which may be a very big advantage of Jupyter. It are often used for data cleaning and transformation, numerical simulation, statistical modelling, machine learning and far more. We use Jupyter in our application to run the algorithm.

3.2 Python

Python is an interpreted, fast, high-level and a general-purpose programming language. Created by Guido van Rossum and it is first released on 1991, Python features a design philosophy that emphasizes code readability, notably using significant whitespace. It's one of the most used programming language presently. It provides constructs that enable clear programming on both small and enormous scales. The K-means used within the system is implemented in Jupyter and the algorithm is written in python language.

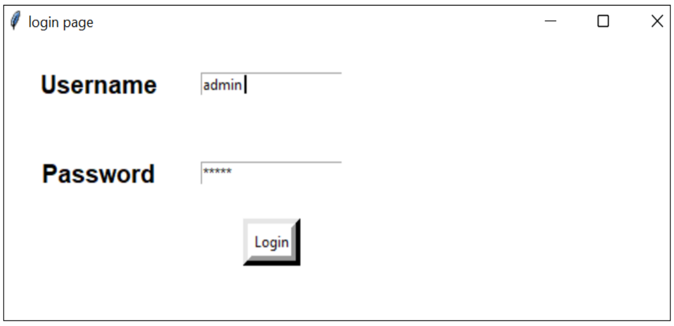
3.3 Tkinter

Tkinter is the standard GUI library for python. Python when combined with Tkinter provides a direct, quick and easy way to create GUI applications. It's the standard python interface to the TK GUI toolkit for UI. It is used for creating user interface in our system.

4 IMPLEMENTATION

The Raw data collected has different attributes and values associated with it. The first and foremost thing to do is find a relation between all these data sets. For example weather data and accident cases data are connected via date. Similarly the road structure and accident cases data are connected by area name. This process starts once the user logged into our system. Once the login is successful, In view he has four different options to choose as functionalities. These are added as additional modules for making our system more informative. The functionalities available are Rules, Risk Prediction, Plot Graph, Entry New Data. Once the user clicks on Rules, A new window opens with two placeholders named support and confidence. If provided correct values it shows the frequent accident occurring patterns in accident data. The second feature, Risk Prediction is the primary module of the application. Once user clicks on Risk Prediction, there will be a button named "TRAIN" which starts the training for the model. Once the model is ready it's shown by a message "Finished clustering using K-Means". It might take up some minutes to get the model ready. Once the model is ready you can choose the areas present in the drop down choice. The places provided belongs to the metropolitical city Bangalore. Once submitted, the model now take the selected city and starts the classification based on clusters by using K-Means. The model finally predicts whether the selected area is "HIGH" or "LOW" accidental prone zone. The third functionality additionally added is "Plot Graph". This presents a dropdown to select a state from list of states of country India. The collected data of these state wise accident are plotted in four different ways. Accident cause, weather condition, Number of Victims and yearly deaths. These data are static and different from the data collected for city Bangalore. The last feature is "Entry new data" which gives user the ability to add new accident occurred. This data is collected and added to the main datasets to make our model further strong and accurate to real time.

5. RESULTS



Username admin

Password *****

Fig 5.1 Log In Page

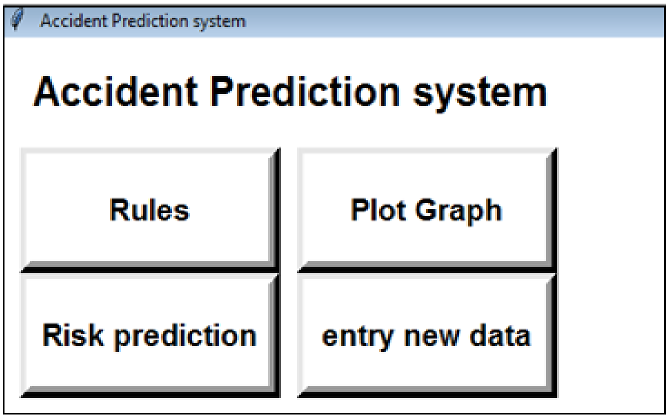
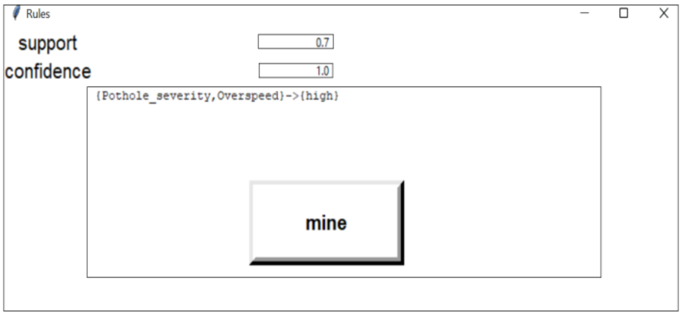


Fig 5.2 Accident Prediction System

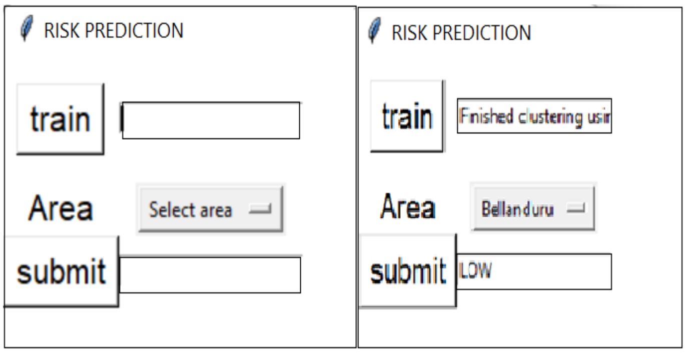


support 0.7

confidence 1.0

```
{Pothole_severity,Overspeed}->{high}
```

Fig 5.3 Rule Mining



Area: Select area

Finished clustering usir

Area: Bellanduru

LOW

Fig 5.4 Risk Prediction Before and After Training

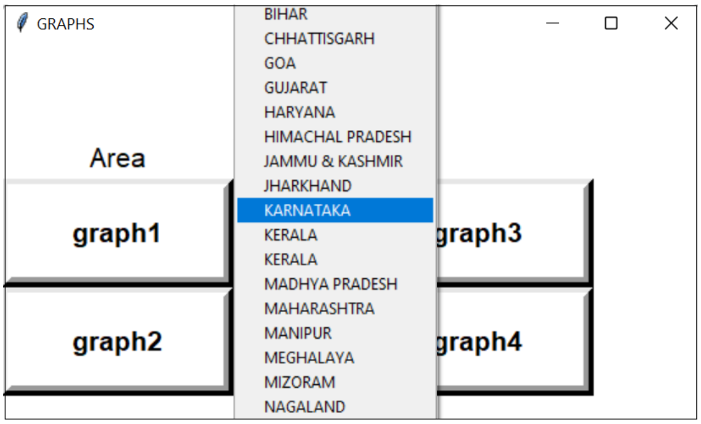


Fig 5.5 Graphs with states dropdown to select

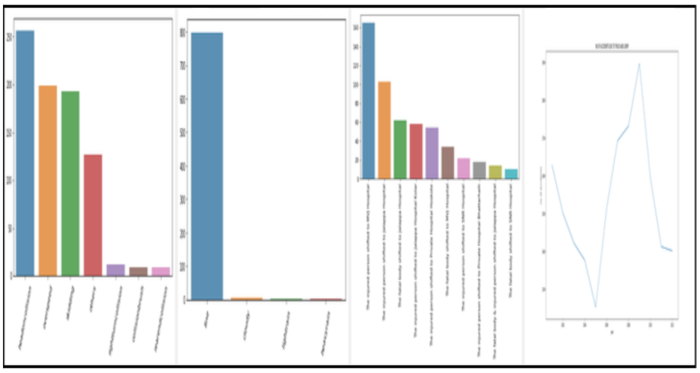


Fig 5.6 Graphs for state karnataka

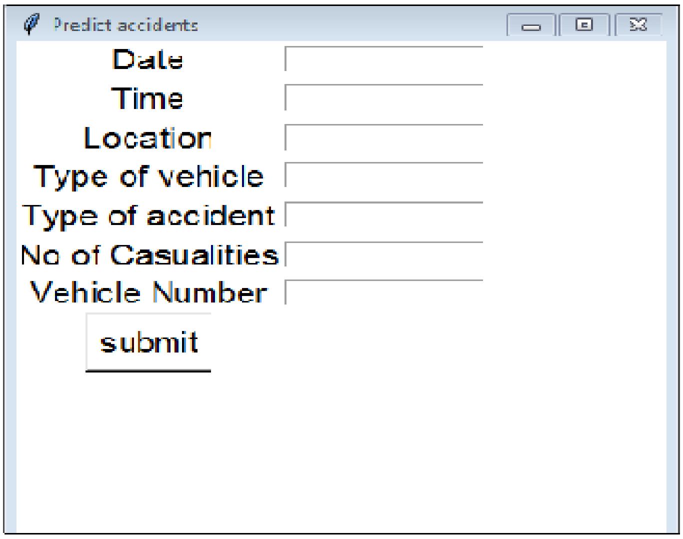


Fig 5.7 Entry New Data

6. CONCLUSIONS

The road accident prediction system aims to develop an application to predict whether the given area in Bangalore city is high accident prone or low accident prone. Road Accidents are caused by various factors. Road Accident cases are hugely affected by the factors such as types of vehicles, pothole severity, overspeed, weather condition, road structure and so on. Using the above factors as attributes we have built an application which gives efficient prediction of road accidents based on the above mentioned factors. Additionally the application provides rule mining which gives the frequent appeared attributes in accident cases. The application also provides different Graphs based on state in India. There is also a used form where new accident cases can be added to the application for updated model.

REFERENCES

[1] Vipul Rana, Hemant Joshi, Deepak Parmar, Pradnya Jadhav, Monika Kanojiya, Road Accident Prediction using Machine Learning Algorithm, IRJET, 2019

[2] Ayushi Jain, Garima Ahuja, Anuranjana, Deepti Mehrotra, Data Mining Approach to Analyze the Road Accidents in India, proc of ICRTIO, 2016.

[3] Baye Atnafu, Gagandeep Kaur, Survey on Analysis and Prediction of Road Traffic Accident Severity Levels using Data Mining Techniques in Maharashtra, India, International journal of current engineering and technology, 2017.

[4] Dinesh Singh, Chalavadi Krishna Mohan, Deep Spatio-Temporal Representation for Detection of Road Accidents Using Stacked Autoencoder, IEEE Transactions on Intelligent Transportation Systems, 2016.

[5] MING ZHENG, TONG LI, RUI ZHU, JING CHEN, Traffic accident's severity prediction: a deep-learning approach based CNN network, IEEE Access, 2017.

[6] Helen W R, N Almelu, S Nivethitha, Mining Road Accident data based on Diverted Attention of Drivers, proc of ICICCS, 2018.

[7] Irina Makarova, Ksenia Shubenkova, Eduard Mukhametdinov, and Anton Pashkevich, Safety related problems of transport system and their solutions, proc of IEEE, 2018.

[8] WHO, "Global status report on road safety 2015" 8.Peden, World Health Organization. Ed. by Margie-2004. World report on road traffic injury prevention. Geneva: World Health Organization.

[9] M. Chang, L. Y., &amp; Chen, W. C. "Data mining of tree- based models to analyse freeway accident frequencies". Journal of Safety Research, 36(4), 365-375

[10] Maze, T. H., Agarwai, M., & Burchett, G. "Whether weather factors matters to traffic demand, traffic safety, and traffic operations and flow".

[11] Teseema, T. B., Abraham, A., & Grosan, C. (2015). "Rule mining and classification of road traffic accidents using adaptive regression trees". International Journal of Simulation, 6(10), 80-94.