ROAD ACCIDENT SEVERITY PREDICTION

Report submitted in partial fulfillment of requirements for
The award of the degree of

BACHELOR OF TECHNOLOGY
In
Computer Science and Engineering

Submitted by

KAMIREDDY PRASANTHI,121710319024
MYLA SUKESH,121710319034
CH ANI SRK,121710319008
G.BHARGAV,121710319014

Under the esteemed guidance of
Mrs.G.KARTHIKA
Assistant Professor, CSE, GIT

Department Of Computer Science and Engineering
GITAM INSTITUTE OF TECHNOLOGY

GITAM

(Deemed to be UNIVERSITY)

VISAKHAPATNAM-530045

2020-2021

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
GITAM INSTITUTE OF TECHNOLOGY
GITAM (Deemed to be University)
VISAKHAPATNAM

DECLARATION

We hereby declare that the project report entitled "**Road Accident Severity Prediction**" is an original work done and provide this opportunity by the Department of Computer Science and Engineering, GITAM Institute of Technology, GITAM (Deemed to be University), Visakhapatnam submitted in partial fulfillment of the requirements for the award of the degree of B.Tech in Computer Science and Engineering. The work has not been submitted to any other college or University to recognize any degree or diploma.

Regd.No	Name	Signature
121710319024	K.Prasanthi	K.Prasanthi
121710319034	M.Sukesh	M.Sukesh
121710319008	Ch. Ani SRK	Ch.AniSRK
121710319014	G.Bhargav	G.Bhargav

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

GITAM INSTITUTE OF TECHNOLOGY

GITAM (Deemed to be University)

VISAKHAPATNAM

BONAFIDE CERTIFICATE

This is to certify that the project report entitled **Road Accident Severity Prediction** is a bonafide record of work carried out by **K.PRASANTHI (121710319024), M.SUKESH (121710319034), CH.ANI SRK (121710319008), G.BHARGAV (121710319014)** submitted in partial fulfillment of the requirement for the award of the degree of Bachelors of Technology in Computer Science and Engineering.

PROJECT GUIDE	**HEAD OF THE DEPARTMENT**
G.KARTHIKA	**Dr. R.SIREESHA**
(ASST. PROFESSOR)	**(PROFESSOR)**
CSE, GIT	**CSE, GIT**
GITAM	**GITAM**

ACKNOWLEDGEMENT

We would like to thank our project guide **G.KARTHIKA**, Asst. Professor, Department of CSE for her stimulating guidance and profuse assistance. We shall always cherish our association with her for her encouragement and valuable suggestions throughout the progress of this work. We consider it a great privilege to work under her guidance and constant support.

We also express our thanks to the project reviewers, **Dr. G.Appa Rao,** Professor, **Dr. M. Padmaja,** Assistant Professor, and **Dr. Srinivas Prasad,** Professor, Department of CSE, GITAM (Deemed to be University), for their valuable suggestions and guidance for doing our project.

We consider it a privilege to express our deepest gratitude to **Dr. R.Sireesha**, Head of the Department, Computer Science Engineering, for her valuable suggestions and constant motivation that immensely helped us complete this project successfully.

Our sincere thanks to **Dr. C. Dharma Raj,** Principal, GITAM Institute of Technology, GITAM (Deemed to be University), for inspiring us to learn new technologies and tools.

Finally, we deem it a great pleasure to thank one and all that helped us directly and indirectly throughout this project.

121710319024	K.Prasanthi
121710319034	M.Sukesh
121710319008	Ch.AniSRK
121710319014	G.Bhargav

TABLE OF CONTENTS

1.ABSTRACT

Traffic accidents are one of the world's most serious problems, resulting in numerous lives, injuries, fatalities, and economic losses every year. Transportation systems must develop accurate models to forecast the severity of traffic accidents. This work builds models to identify a collection of relevant characteristics and construct a model for categorising injury severity. These models are created using a variety of machine learning approaches. On traffic accident data, supervised machine learning techniques and unsupervised machine learning algorithms are used. Using decision tree analysis and Bernoulli naive Bayes classification methods, this study examines an accident data set that includes road conditions and vehicular features.

Our project is divided into four sections. The first module is concerned with populating the database with information about road conditions and vehicular factors. The second module focuses on developing the necessary graphical user interfaces for project management. The dataset is then classified using Decision Trees in the third module.The fourth module involves calculating the accuracies and generating a comparison chart using Bernoulli Naive Bayes Classification on the same dataset.

2.INTRODUCTION

Traffic accidents are a major cause of deaths, injuries, and property damage, as well as a serious public health and traffic safety problem. Accidents contribute significantly to traffic congestion and delays. Accident management is critical for reducing accident effects, enhancing traffic safety, and increasing transportation system efficiency. As a result, two important processes, severity prediction and length estimate, are critical. Emergency responders can use accurate severity and duration projections to assess the severity of incidents, anticipate probable consequences, and conduct effective accident control techniques.

When a road accident happens, various factors such as road conditions, weather conditions, and other factors influence the severity of the accident. The examination of these incidents aids in determining the characteristics that contribute to the accident's criticality. Road accident analysis is carried out in this project utilising machine learning algorithm approaches, which allow the elements that influence the severity of the accident to be identified.

We applied classification algorithms to determine the relationship between numerous factors that contribute to the accident's criticality. To begin, the criticality of accidents for the given attribute values is determined using the Nave Bayes classification technique. The Decision Tree method is then employed to carry out the same task. The accuracy of both algorithms is compared in order to determine which is the better algorithm. The goal of this study is to find a relationship between the elements that cause an accident and the severity of the accident. It determines if an accident is critical or non-critical based on the specified parameters.

3.LITERATURE SURVEY

Severity and duration have long been prominent study subjects in accident analysis as two major elements. The majority of prior research only looked at one of the two factors: severity and duration. In the case of severity analysis, for example,

- Using Washington State accident data, Chang and Mannering [1] investigated the association between the severity of injuries and the severity of vehicles.
- Manner and Wünsch-Ziegler [2] investigated the risk factors and found significant results for related factors.
- In terms of time, Chung [3] used highway accident data obtained in Korea to simulate accident duration.
- Anastasopoulos et al. [4] introduced a Bayesian network model that can learn emerging trends and estimate the time it takes for an accident to be cleared. Nonetheless, several researches discovered that the intensity of the injury influences the length time.
- For example, Nam and Mannering [5] discovered that whether there is a death or injury in an accident affects the length of the accident. Furthermore, as illustrated in Figure 1, the severity prediction and duration estimation operations in the accident management system are linked operations. As a result, the two indications should be assessed in tandem and integrated into a single model system.

Most previous scholars explored severity analysis as a single comprehensive indication, which comprises primarily three aspects: number of fatalities, number of injuries, and property damage; for example,

- Manner and Wünsch-Ziegler [2]considered accident severity as a single independent variable, with four options: death, serious injury, mild injury, and property damage.
- Property damage alone, probable harm, and injury were defined by Milton et al. [6].
- Malyshkina and Mannering [7] calculated severity using only three options: death, injuries, and property damage. Furthermore, several study looked at only one or two of the three severity factors.

- For example, Stone and Broughton [8] and Sze and Wong [9] focused solely on mortality, dividing accidents into two categories: fatal and nonfatal. No injury, prospective injury, non-incapacitating, incapacitating, and mortality were the categories of injury severity outlined by Delen et al. [10]. Similarly, Ballesteros et al. [11] and Roudsari et al. [12] only looked at the number of deaths and injuries, but not the amount of property damage. Varied sorts of losses and the quantity of failures result in different reaction measures, which might endure for a long period. For example, in Zhang's research [13], a level 2 accident is defined as one that causes $167–5000 in property damage or one that causes 1–3 injuries.

As previously stated, most earlier research looked at accident severity and length independently, even though they were proven to be related. Furthermore, earlier research only looked at one or two of the three components of accident severity: the number of fatalities, injuries, and property damage. As a result, the current project intends to create a model system that can evaluate both the severity and length of an accident. In addition, three accident severity indicators, reflecting the number of fatalities, injuries, and property damage, will be evaluated.

4.IDENTIFY THE PROBLEM STATEMENT

The current system is manual, where the government sector uses ledger data and analyzes the data manually. Based on the analysis, they will take precautionary measures to reduce traffic accidents and related injuries. We also get many tools and software to maintain traffic accidents and related injuries, these tools just collect the data stores in sever, but no analysis is done. The current system is a manual process to maintain traffic accident data and injury severity using books or ledgers. Discovering the associations among the road conditions and vehicular aspects is the key factor in reducing traffic accidents.

5.PROPOSED MODEL

In this project, we are predicting the accident severity using Decision Tree Classification and Naïve Bayes Classification,

DECISION TREE CLASSIFICATION

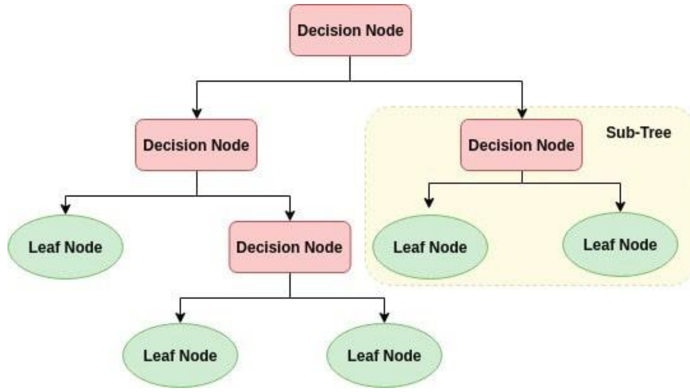
For a tree structure, a decision tree constructs classification or regression models. In addition to building a connected decision tree, it divides the data collection into smaller subsets. The results are also a tree of decision nodes and leaf nodes, with the highest decision node being the root node. Categorical and numerical data can be handled by the decision trees. This tree-like layout is built on this algorithm. Since the marked data is provided beforehand, the learning technique is supervised, and the output depends on the data previously classed. In this algorithm there are two types of nodes: Node of decision; Node of leaf. The Leaf Node is the last output node. This is the last node of the tree not separable and without branches. The best attribute is chosen as a root node by several ways. Information gain is one method for evaluating the root node from different other attributes. In a decision-tab classification algorithm, the sum of information gain is always maximized and the attribute with the maximal information gain is divided first. The equation is used to calculate the gain is:

$$Gain(S, A) \equiv Entropy(S) - \sum_{v \in D_A} \frac{|S_v|}{|S|} Entropy(S_v)$$

In a certain feature, Entropy determines the impurity. It says the data are random. Entropy of the same equation is determined as:

$$Entropy\ (p) \ = \ - \sum_{i=1}^{N} p_i \ log_2\, p_i$$

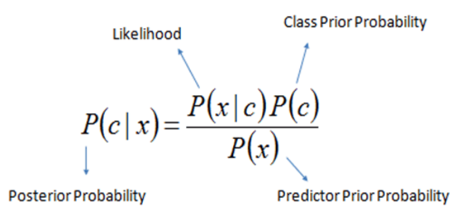
Where N= number of possible categories.



The decision tree can be structured for the optimal output until the root node is chosen.

NAÏVE BAYES CLASSIFICATION:

It is essentially based on the theorem and the assumption of predictor freedom. It follows the classification method. In its simplest form, the classification of Naive Bayes means that one function in a class is not connected to another. The Naive Bayes model is easy to create and especially useful for larger sets of data. Due to its simplicity, Naive Bayes is often regarded as the most sophisticated methods of classification.



$$P(c\,|\,X) = P(x_1\,|\,c)\times P(x_2\,|\,c)\times\cdots\times P(x_n\,|\,c)\times P(c)$$

The terms reflect the following in the above formulation:

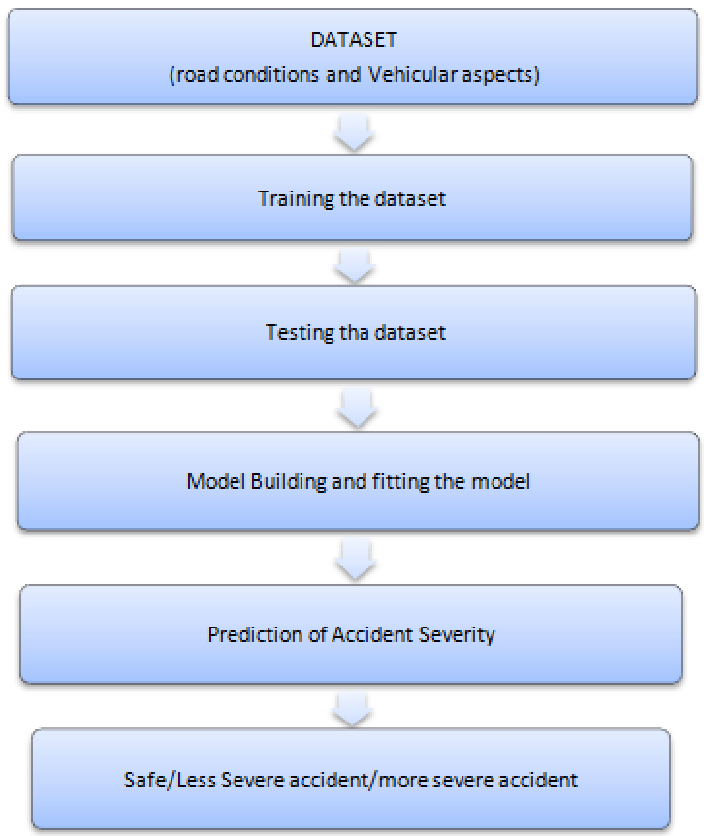
- The class (c, target) predictor's retrospective likelihood is P(c|x) (x, attributes).
- P(c) is the probability of class precedence.
- The expectation is P(x|c), which is the likelihood of a class predictor.
- The prior probability of P(x) is the predictor.

It is easy and fast to predict the test data set class. It is also useful for the prediction of several classes. When the independence condition is fulfilled, a Naive Bayes classifier executes other models such as logistic regressions which needs fewer training data. Compared to a numerical variable, categorical input variables perform well. If the categorical variable in the test data set has a class that does not match the actual data set, the model would offer a 0 (zero) chance and cannot estimate. This is sometimes called the "Null Frequency." Another failure is the assumption of separate predictors.In real life, it is almost difficult to obtain a set of entirely independent predictors.

6.SYSTEM ARCHITECTURE

A System Architecture is a conceptual model that specifies a system's structure, behaviour, and other aspects. A formal description and representation of a system arranged in a way that facilitates reasoning about the system's structures and behaviours is known as an architectural description.

The System Architecture of our Project is as follows:



7.DATASET SPECIFICATIONS

A dataset is a collection of data or examples that have been gathered and organised in a certain way. It might be stored in a database or an array. It may be used to either train or test a model. Data in a dataset might be numerical, category, or ordinal. To train the model, which will yield the prediction model, a training dataset is used. To check the correctness of the data, a testing model is utilised.

- A dataset is taken from Kaggle, which contains 5587 rows and 19 columns.
- This dataset is the combination of road conditions and vehicular aspects attributes, which includes the road curvature, slanted, visibility, vision obstructers, and presence of the hairpin bends the state of the vehicle, which is reflected by the age of the vehicle, engine capacity, propulsion type, and transmission type, etc.
- The output column accident severity has the value "0", "1", "2".

8.SYSTEM REQUIREMENTS

SOFTWARE REQUIREMENTS

- It is necessary to have a 64-bit Ubuntu or Windows operating system.
- The user interface is designed with Python Qt Designer.
- SQLite3 is used to store database entities.
- The layout-designed user interface (UI) is converted to python code using Pyuic.
- Python is a programming language used for coding.

HARDWARE REQUIREMENTS

The conditions for the project's execution are specified in the hardware requirements. This hardware must be available in order for the project to be completed successfully.

- A CPU with a clock speed of at least 2.16 GHz is required.
- A minimum of 4 GB RAM is required.
- It necessitates the use of a 64-bit architecture.
- It requires a minimum of 500GB of storage.

9.PROGRAMMING LANGUAGES

Python is the programming language utilised in this project. Python is a dynamically semantic, interpreted, object-oriented high-level programming language. The combination of high-level built-in data structures, dynamic typing, and dynamic binding makes it ideal for Rapid Application Development and usage as a scripting or glue language to link existing components. Python's basic, easy-to-learn syntax prioritises readability, lowering software maintenance costs. Modules and packages are supported by Python, which fosters programme modularity and code reuse. The Python interpreter and its substantial standard library are free to download and distribute in source or binary form for all major platforms.

The following are some of the packages that were utilised in this project:

- Numpy: This module is the cornerstone of Python's scientific computing capabilities.
- Pandas: This is a data analysis module. With back-end source code written entirely in C or Python, it gives highly optimised performance.
- Sys: The sys module in Python has a number of methods and variables for manipulating various aspects of the Python runtime environment.

10.FEASIBLE STUDY

A feasibility study is a simplified version of the System Analysis and Design Process. The investigation begins with a classification of the problem definition. The purpose of feasibility is to see if something is worth trying. The analyst creates a logical model of the system after creating an acceptance problem specification. The hunt for alternatives is thoroughly examined.

Economic Feasibility:

This evaluation usually includes a cost-benefit analysis of the project, which aids firms in determining the project's viability, cost, and benefits before allocating financial resources. It also functions as an impartial project assessment and boosts project credibility by assisting decision-makers in determining the proposed project's positive economic advantages to the organization. Our project is economically feasible since we employed open-source software such as "UBUNTU," "PYTHON," the "PYQT" designer tool, and "PYUIC."

Technical Feasibility:

The focus of this evaluation is on the organization's technological resources. It assists organisations in determining if technical resources are enough for their needs. The hardware, software, and other technological needs of the proposed system are also evaluated for technical viability. To ensure technological feasibility, a tool prototype was created.

11.UML DIAGRAMS

UML is an organized demonstration language that includes an interconnected range of diagrams created to help programmers characterize, illustrate, schedule and document programmes and market show frameworks. and other non-programming framing mechanisms. The UML is a collection of the best design practices collected for a while. The UML offers an overview of best programming practices for showing huge and nuanced structures. The UML is an essential component of the production of articles and the measurement for product development.UML essentially uses graphical documentation to express the programming schedule. The UML supports project groups to carry out research and to approve the structural plans of the product.

The UML diagrams designed for this project are:

- Sequence diagram
- State chart diagram
- Use case diagram

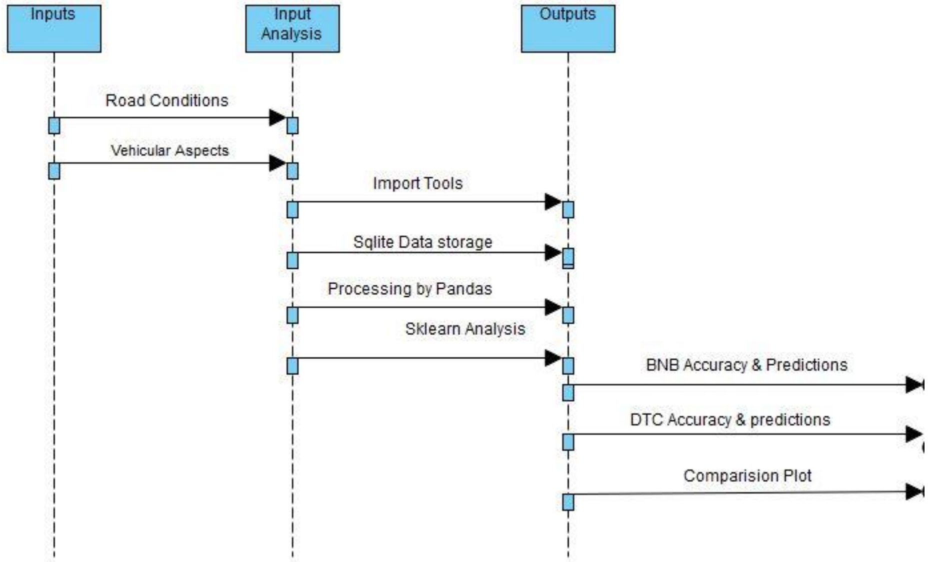
SEQUENCE DIAGRAM: Interaction diagrams such as UML sequence diagrams show how they work. They concentrate on the relationship between objects in a collaborative setting. Sequence Diagram is time-sensitive and physically displays the correspondence order using the diagram's vertical line to show the time and the messages sent and received.

Sequence Diagrams show:

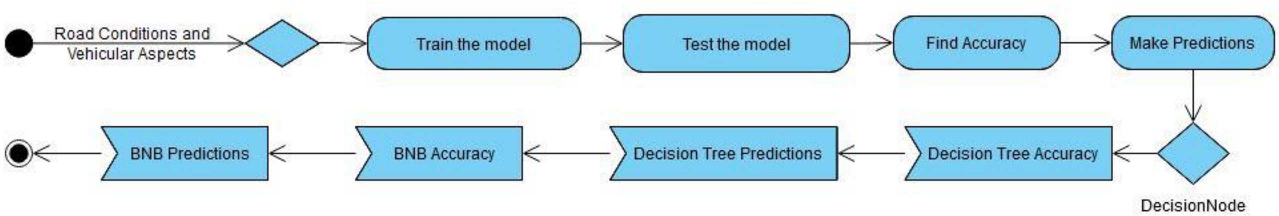
- Exchanges that take place in collaboration between the customer and the machine, between systems or between subsystems.
- Interactions in a significant degree(sometimes referred to as system sequence diagrams)

The message passes from the user's interface to the forecasting machine. Among them there are six letters.

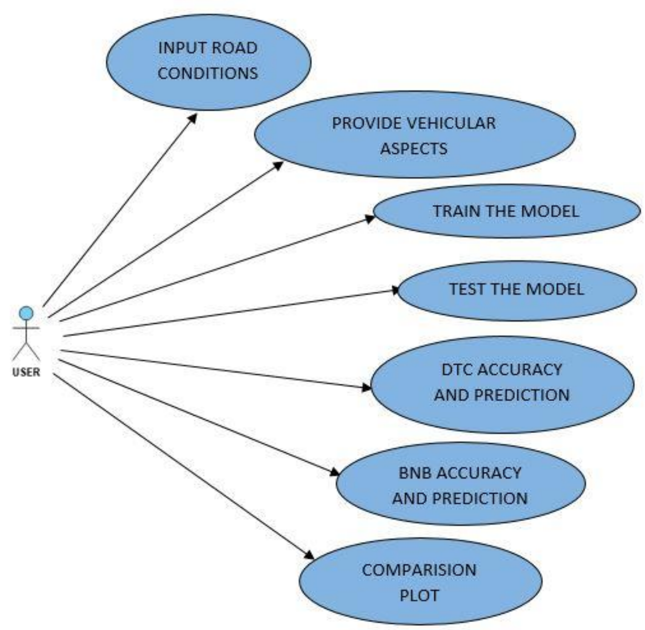
The following chart shows these:



ACTIVITY DIAGRAM: Another significant diagram in UML for describing dynamic characteristics of the system is the activity diagram. An activity diagram is a flow chart that depicts the transition from one activity to the next. The action can be defined as a system operation. As a result, the control flow is transferred from one operation to the next. The activity diagram shows that the initial step is to give the dataset, which includes Road Conditions and Vehicular Aspects, followed by training the model, testing the model, and calculating the accuracies, as well as BNB and DTC predictions.



USE CASE DIAGRAM: A case diagram of UML use is the critical kind of framework/programming particularly for a missing schedule. The behaviour in usage cases is specified, but the precise way to implement the behaviour is not defined. Once developed, the use cases can be shown textually and visually. The core theory of case modelling is that it allows the end user to build a device. It is the easiest way to explain the behaviour of the system to users by characterizing all freely perceptible framework actions in their own language.The user can perform the operations like providing Road Conditions, Vehicular Aspects and then calculating the accuracies of BNB & DTC along with the comparison plot.
The actors are the user and machine learning model in this situation. The system is defined as the mechanism for Accident Severity.



12.IMPLEMENTATION OF GUI

1.IMPORTING THE NECESSARY LIBRARIES:

- Pandas: Reading the dataset saved with an extension of .csv.
- Numpy: Performing various mathematical operations.
- Sklearn: For importing the various algorithms and accuracy metrics.

In [1]:
```
import numpy as np
import pandas as pd
from sklearn import *
from sklearn.tree import DecisionTreeClassifier
from sklearn.naive_bayes import BernoulliNB
from sklearn.metrics import accuracy_score
```

2.GATHERING THE RELEVANT DATASET:

For any Machine Learning model, gathering a relevant dataset is the most important task to be performed. If the dataset chosen is appropriate, then the model can be easily trained. This particular dataset contains 5586 rows and 19 columns.

In [2]:
```
training_data = np.genfromtxt('acset1.csv', delimiter=',')
```

In [3]:
```
pd.read_csv("acset1.csv")
```

Out[3]:

	646.04	9	121.04	1	1.1	59	2.17	1.2	0.2	1.3	156.6	100	7	44	6	1533	1.4	0	1.5
0	629.85	6	115.07	1	1	57	2.90	1	0.1000	1	163.88	100	9	41	9	1496	2	1	1
1	858.96	10	97.73	1	1	41	2.74	1	0.2400	1	132.49	100	6	44	10	1765	1	0	1
2	402.65	4	17.81	0	0	93	0.06	0	0.0141	0	85.74	60	2	80	22	819	0	0	0
3	934.74	8	65.00	1	1	47	1.70	1	0.1300	1	109.77	100	6	47	19	1401	2	1	1
4	638.66	8	117.46	1	1	46	2.30	1	0.2400	1	128.04	100	8	39	17	1788	1	0	1
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
5581	1146.48	28	191.13	2	2	22	3.51	2	0.2600	2	240.85	200	10	34	1	2075	1	1	0
5582	346.39	0	12.94	0	0	98	0.15	0	0.0476	0	50.65	60	4	58	48	1052	0	0	1
5583	1433.71	35	184.17	2	2	37	3.61	2	0.3300	2	211.95	200	18	18	4	2560	2	1	0
5584	781.63	9	34.76	1	1	49	1.12	1	0.0800	1	165.60	100	9	40	10	1481	1	1	0
5585	631.43	8	31.86	1	1	60	1.45	1	0.2500	1	193.67	100	7	38	12	1703	1	1	0

5586 rows × 19 columns

3.SPLITTING THE DATASET INTO TRAIN AND TEST:

The data can be split into two categories: "train" and "test". Training details are taken into account for model training, and test results are taken into account for estimation and accuracy measurement. The computer will learn more and forecast the right outcome with additional training data. The division is made on the undertaken results, since it gives a more precise forecast.

```
In [4]:  inputs = training_data[:,:-1]

In [5]:  outputs = training_data[:, -1]

In [6]:  training_inputs = inputs[:3000]

In [7]:  training_outputs = outputs[:3000]

In [8]:  testing_inputs = inputs[3000:]

In [9]:  testing_outputs = outputs[3000:]
```

4.BUILDING THE MODEL:

The model was developed with a variety of classification algorithms present in machine learning after the data were separated into train and test data. They include the classification of Decision trees and Naive Bayes.

<u>DECISION TREE CLASSIFICATION:</u>

Assumptions in Decision Tree Classification:

- The root is considered as the initial training set.
- The data are assumed to be categorical to quantify the gain in detail.
- Recursive allocation of records is carried out based on the attribute values obtained.
- For ordering attributes as root or an internal node, a statistical approach is used.

```
In [10]: classifier = DecisionTreeClassifier()

In [11]: classifier.fit(training_inputs, training_outputs)
Out[11]: DecisionTreeClassifier()

In [12]: predictions = classifier.predict(testing_inputs)

In [13]: print(predictions)
         [1. 0. 1. ... 2. 1. 1.]

In [14]: accuracy = 100.0 * accuracy_score(testing_outputs, predictions)

In [15]: print ("The accuracy of DT Classifier on testing data is: " + str(accuracy))
         The accuracy of DT Classifier on testing data is: 99.45883262466178
```

NAÏVE BAYES CLASSIFICATION:

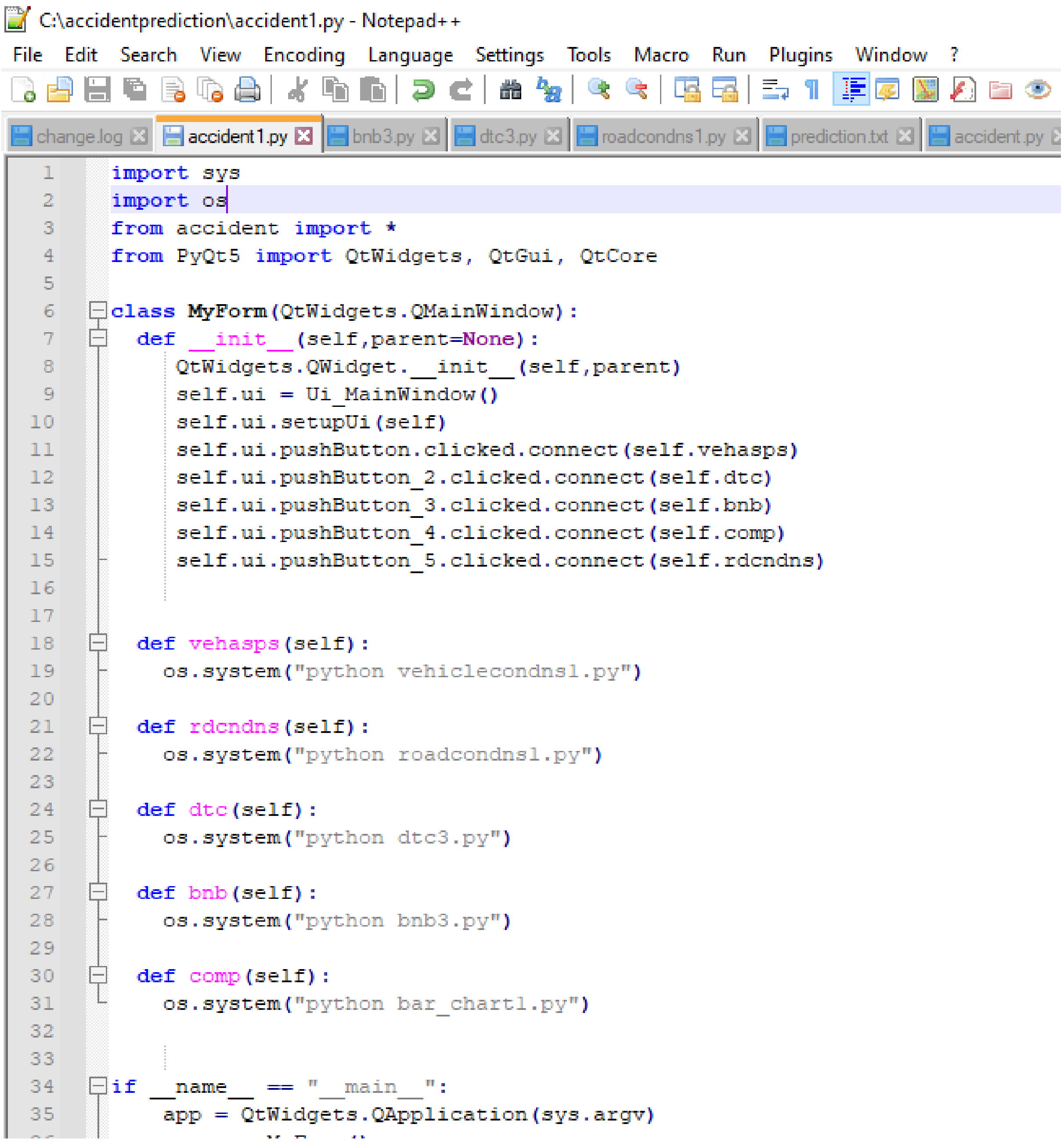
Steps performed in K-Nearest Neighbors Classification:

- Separation of the class is done.
- Dataset summarization.
- Data summarization based on theclass.
- Calculate the density function for Gaussian probability.
- Calculate the class probabilities.

```
In [10]: classifier = BernoulliNB()

In [11]: classifier.fit(training_inputs, training_outputs)
Out[11]: BernoulliNB()

In [12]: predictions = classifier.predict(testing_inputs)

In [13]: print(predictions)
         [1. 0. 1. ... 2. 2. 2.]

In [14]: accuracy = 100.0 * accuracy_score(testing_outputs, predictions)

In [15]: print ("The accuracy of BNB Classifier on testing data is: " + str(accuracy))
         The accuracy of BNB Classifier on testing data is: 83.41708542713567
```

5.DEPLOYING THE MODEL:

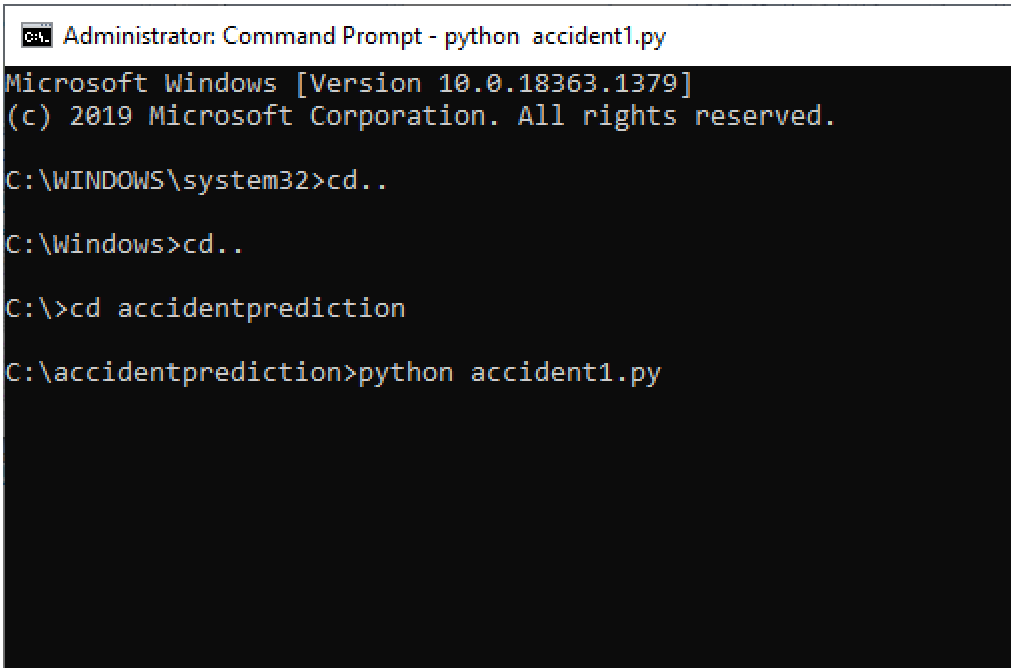
The model, after training, is ready to get deployed. For deploying the model, we use the PYQT5 tool, a python framework, and can be integrated with web applications to create graphical user interfaces.

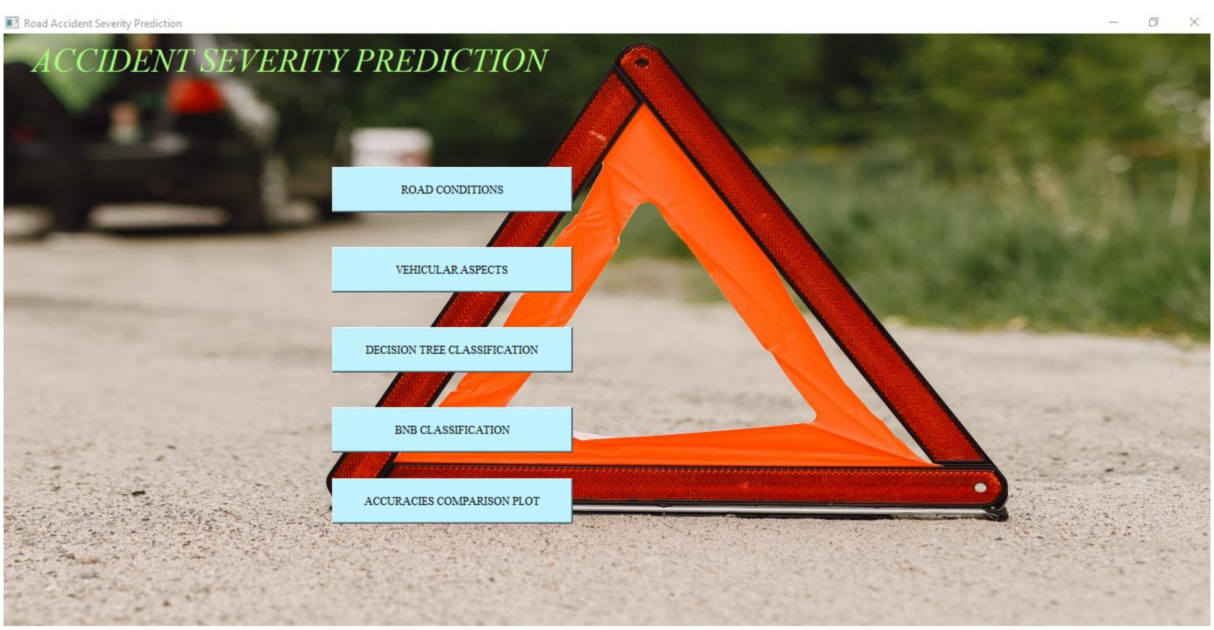



```python
import sys
import os
from accident import *
from PyQt5 import QtWidgets, QtGui, QtCore

class MyForm(QtWidgets.QMainWindow):
  def __init__(self,parent=None):
     QtWidgets.QWidget.__init__(self,parent)
     self.ui = Ui_MainWindow()
     self.ui.setupUi(self)
     self.ui.pushButton.clicked.connect(self.vehasps)
     self.ui.pushButton_2.clicked.connect(self.dtc)
     self.ui.pushButton_3.clicked.connect(self.bnb)
     self.ui.pushButton_4.clicked.connect(self.comp)
     self.ui.pushButton_5.clicked.connect(self.rdcndns)


  def vehasps(self):
    os.system("python vehiclecondns1.py")

  def rdcndns(self):
    os.system("python roadcondns1.py")

  def dtc(self):
    os.system("python dtc3.py")

  def bnb(self):
    os.system("python bnb3.py")

  def comp(self):
    os.system("python bar_chart1.py")


if __name__ == "__main__":
    app = QtWidgets.QApplication(sys.argv)
```

13.RESULTS

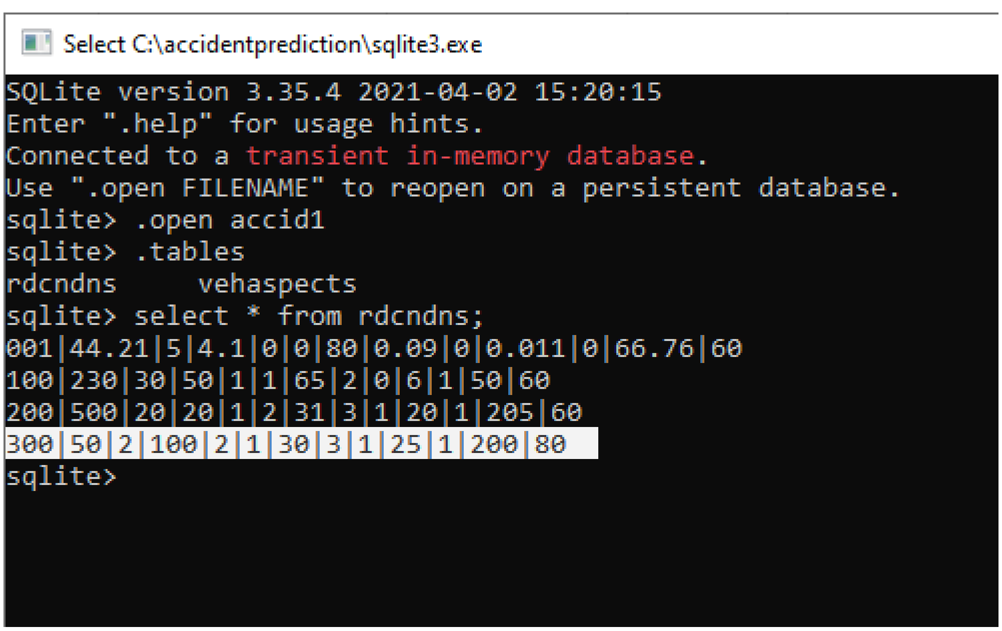
Running the application in command prompt,



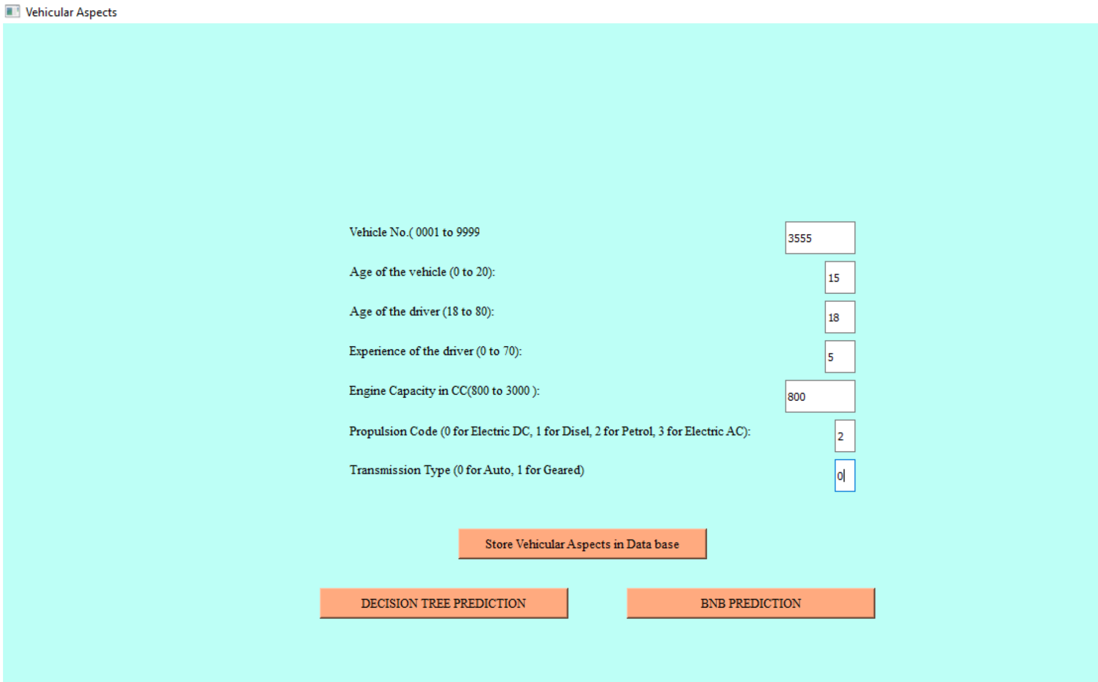


Road conditions:

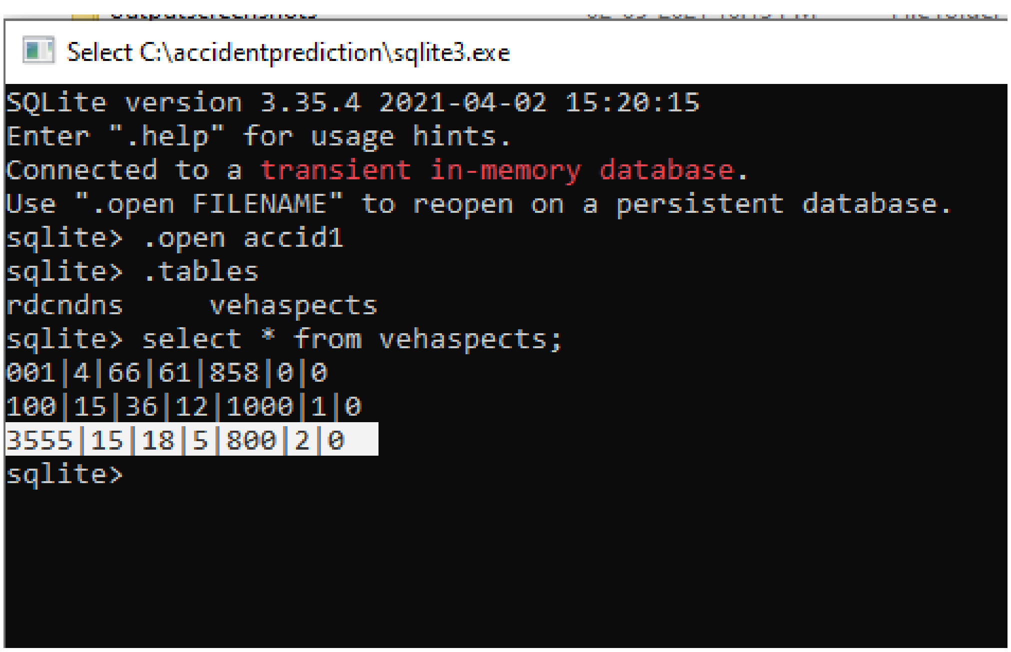
The values of road conditions are stored in the database,



Vehicular Aspects:



The values of Vehicular Aspects are stored in the database,



DECISION TREE CLASSIFICATION:

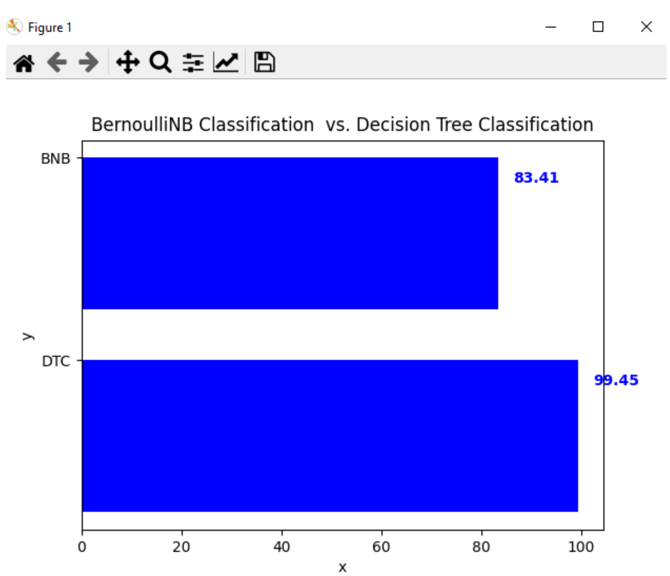
Administrator: Command Prompt

```
Microsoft Windows [Version 10.0.18363.1379]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\WINDOWS\system32>cd../..

C:\>cd accidentprediction

C:\accidentprediction>python accident1.py
The accuracy of DT Classifier on testing data is: 99.45883262466178
DT prediction on the first test set is: [0.]
DT prediction on the second test set is: [1.]
DT prediction on the third test set is: [2.]
```

NAÏVE BAYES CLASSIFCATION:

```
C:\accidentprediction>python accident1.py
The accuracy of BNB Classifier on testing data is: 83.41708542713567
BNB prediction on the first test set is: [0.]
BNB prediction on the second test set is: [2.]
BNB prediction on the third test set is: [2.]
```

ACCURACY COMPARISON PLOT:



14.CONCLUSION

This project entitled "Road Accident Severity Prediction" is helpful to the Roads & Transport Departments to identify the accident-prone spots to plan the precautionary measures to prevent accidents. The project is helpful for travelers traveling on Ghat roads to take more care at the accident spots. This study focuses on identifying characteristics that influence the severity of expressway accidents in order to assist responsible authorities in developing accident prevention policies or procedures that successfully mitigate or prevent unforeseen accident losses. The results confirmed that speed on a road segment is the sole significant factor determining the severity of expressway collisions; hence, speed limits on expressways should be imposed for drivers through rigorous expressway speed inspections under regulations.

FUTURE SCOPE:

The project is currently implemented as a stand-alone application. It has to be examined at more to see if it can be made into a web/mobile application. This project may be improved by implementing a data gathering system that keeps track of the closest hospitals that are involved.

15.REFERENCES

[1] Arun Prasath N, Dr.M.Punithavalli: BOUNDED MEMORY BASED FREQUENT PATTERN GROWTH APPROACH WITH DEEP NEURAL NETWORK AND DECISION TREE FOR ROAD ACCIDENT PREDICTION,e-ISSN: 0976-5166

[2] An, S.; Zhang, T; Zhang, X; Wang, J: Unrecorded accidents detection on highways based on temporal data mining, Mathematical Problems in Engineering, 2014, pp.1-7.

[3] Deublein, M.; Schubert, M.; Adey, B. T; Köhler, J; Faber, M. H.: .Prediction of road accidents: A Bayesian hierarchical approach", Accident Analysis & Prevention, 51, pp.274-291.

[4] H. Mannera and L. Wünsch-Ziegler, "Analyzing the severity of accidents on the German Autobahn," *Accident Analysis and Prevention*, vol. 57, no. 8, pp. 40–48, 2013.

[5] H. Y. Zhang, *Analyzing Traffic Accident Situation with Bayesian Network [Doctor Dissertation]*, Jilin University, 2013.

[6] M. Abdel-Aty and K. Haleem, "Analyzing angle crashes at unsignalized intersections using machine learning techniques," *Accident Analysis and Prevention*, vol. 43, no. 1, pp. 461–470, 2011.