A Minor Project Report on
Traffic Accident Severity Detection

Submitted to Manipal University, Jaipur
Towards the partial fulfillment for the Award of the Degree of

BACHELOR OF TECHNOLOGY
In Computer and Communication Engineering
2020-2024

By
Name: Aakrit Bhargava
Registration Number: 209303131

Under the guidance of
Name: G.L Saini

Department of Computer and Communication Engineering
Manipal University Jaipur
Jaipur, Rajasthan

Section 1: Introduction

Motivation

Road accidents are a grave concern for most nations because they can cause severe injuries and fatalities. According to the World Health Organization's Global Status Report, approximately 1.25 million deaths per year are because of road accident injuries, and most fatality rates were in lower income countries [1]. Our motivation is to predict the accident severity of any road, which will play a crucial factor for traffic control authorities to take proactive precautionary measures. In addition, the dataset we chose was rarely solved from a prediction point of view, so we took this opportunity to predict the severity of the accident. Also, we chose this highly imbalanced dataset as we wanted to apply acquired concepts as part of our 3rd year minor project course.

Objective

This project predicts the severity of an accident by training an efficient machine learning model with the help of existing accident data from 1992-2019. This project is majorly focused on predicting rarer classes accurately such as Serious and Fatal.

Section 2: System Design & Implementation details

Algorithms, technologies, and tools

Five classification algorithms were used and evaluated to predict the accident severity. Algorithms considered for classification were K-nearest neighbors, Naive Bayes classifier, Random Forest classifier, Logistic Regression and SVM. The first thing that came to our mind is that severity is based on different decisions like road, weather conditions. So, we tried different decision tree classifiers. We also tried ensemble methods as the data is very imbalanced.

K-Nearest Neighbors Classifier: This model is a non-parametric method used for classification. We tried using different values of neighbors and got the best result for considering three neighbors for each point and weights parameter was set to 'distance' which weights the points by an inverse of their distances. Closer neighbors of a query point will have a greater influence than neighbors which are farther away. This model did perform well on this dataset probably because the data was multidimensional, the points were remarkably close to many data points which were classified in different classes. F1- score of Accident Severity prediction from this model was 0.95.

Naive Bayes Classifier: This model is a probabilistic framework for solving classification problems. Since our features are independent of each other, this model works well to predict the values for a given set of features. As the features selected are discrete, we used multinomial Naive Bayes classifier instead of Gaussian. We tried this algorithm after data cleaning on the entire dataset which gave an F1-score for ‘serious’ severity prediction of 0.04. The same on an under sampled data with an equal ratio of different accident severity gave an ‘serious’ severity prediction F1-score of 0.74

Random Forest classifier: As our data is very imbalanced, we tried ensemble methods, of which random forest classifier is one. A random forest is a meta estimator that fits several decision tree classifiers on various sub-samples of the dataset and uses averaging to improve predictive accuracy and control over-fitting. We tried this algorithm with different class weights such as incremental, multiplicative, exponential weights. All of these gave an awfully bad F1-score for ‘severe’ output class. The inbuilt class weight - balanced subsample, gave us the best results. The F1 scores are .96 for ‘severe’ class and 0.97 overall for the under sampled data

Logistic Regression: This model is the basic and popular for solving classification problems. Unlike Linear Regression, Logistic regression model uses a sigmoid function to deal with outliers. Class weights parameter sets the weights for imbalanced classes by adjusting weights inversely proportional to class frequency. As the dataset was highly imbalanced, even after using ‘balanced’ class weight, the model did not perform well. Using the under-sampling method, the majority class which was the ‘Slight’ accident type was under sampled. On this under sampled data, logistic regression model performed better than using ‘balanced’ class weights. As both the classes were not exactly separable, this model gave the f1-score equals 0.88.

Support Vector Machine: SVMs are based on the idea of finding a hyperplane that best divides a data set into two classes. As the features in this dataset were very sparse and non-separable by any hyperplane, SVM did not work at all for this dataset. For larger dataset, SVM training time is high. When we tried to train the dataset using SVM it ran forever as both the classes were not exactly separable, this model gave the f1-score equals 0.96.

Technologies & Tools used

For developing this project, the tools and technologies have been used.

Python: Python is easy to understand language and has a rich set of libraries to use for data pre-processing, modeling, and evaluating the algorithms. Moreover, python has particularly good community support which is especially useful for debugging the code.

Jupyter Notebook: Jupyter notebook is a simple and interactive tool for running python code. Also, it has many different sets of features such as downloadable to .py, .ipynb, and .html files.

Section 3: Experiments / Proof of concept evaluation

Dataset used:

We have used Road Accident Incidence dataset for our project. The dataset contains 250,000 records and 70 columns including weather conditions, Road class, road type, junction details, road surface conditions, light conditions, etc. Our dataset is very imbalanced with data corresponding to slight severity is 84.84%, serious severity is 13.86% and for fatal severity is 1.30%. For validation of data, we have separated the dataset based on the accident severity and choose data for train and test dataset in equal ratios. We have used k-fold validation with 5 folds from each subset.

Methodology followed

Data Pre-processing techniques: The dataset is inputted by replacing Nan and missing values with the most frequent values of the corresponding column. All the categorical values have been labeled by integers from 0 to n for each column. Time has been converted to categorial feature with 2 values i.e., daytime and nighttime.

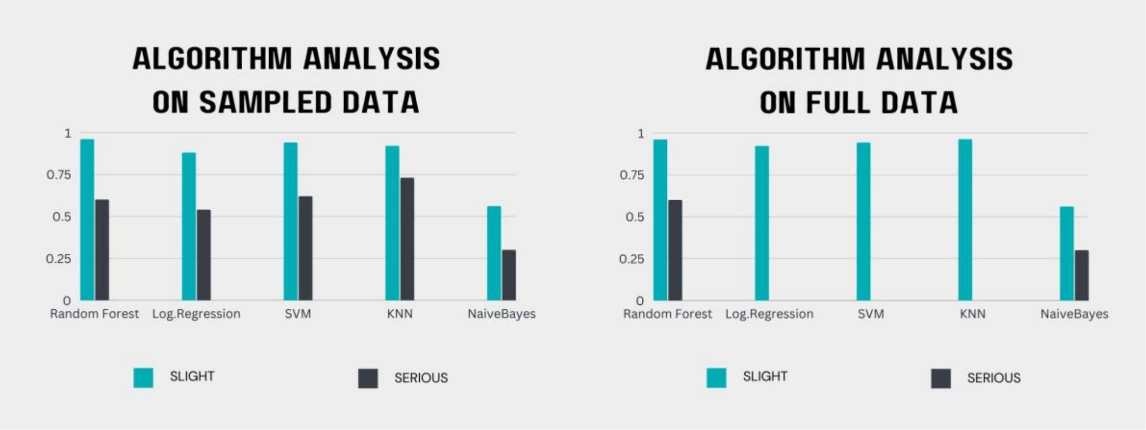
The data is visualized for correlation. Negatively correlated features are selected to be dropped. Feature importance is plotted to visualize and only features with high importance are taken into consideration for predicting accident severity.

The multi class label is converted to binary class by merging “Serious” and “Fatal” to Serious class.

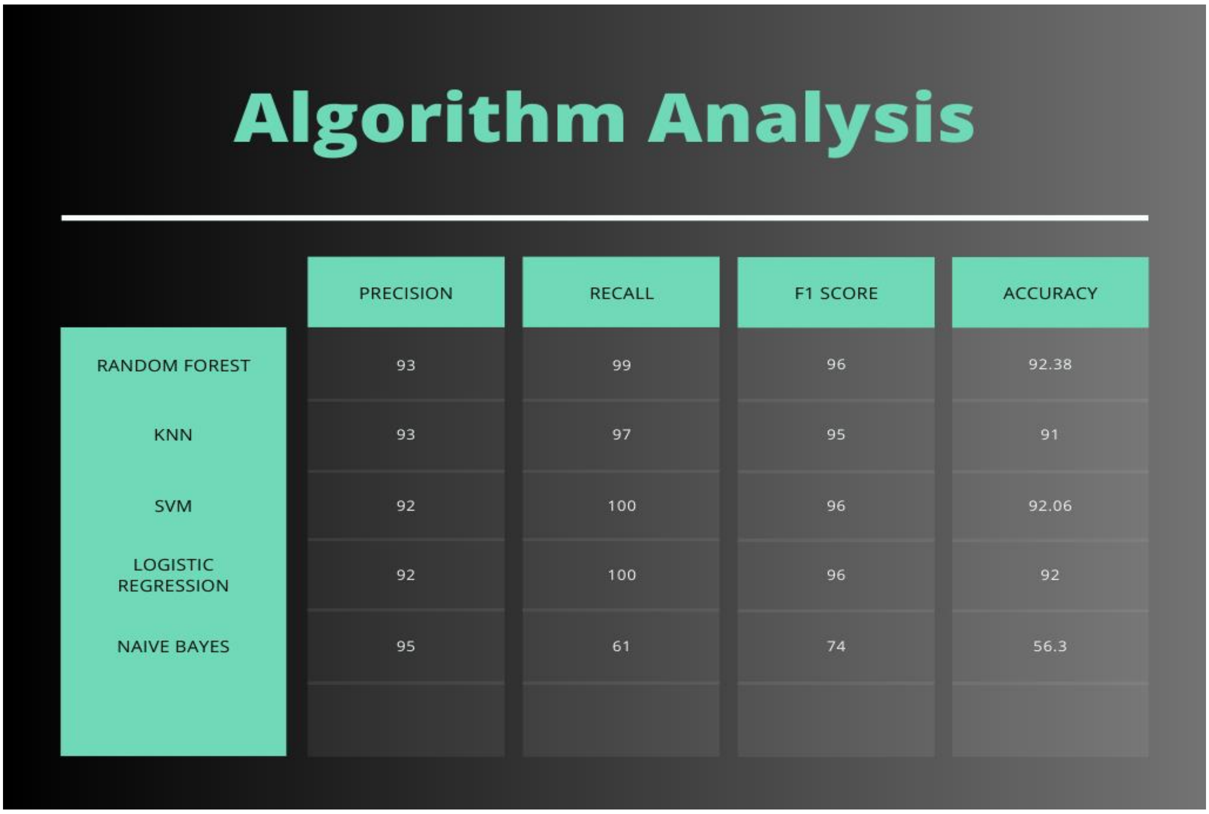
Feature Selection: The dataset has 70 attributes describing the incident of an accident. There are mixed types of data such as continuous and categorical. Manually dropped a few columns due to its inconsistency in values such as Accident ID, and Location ID. For selecting the best features, below functions are used from sklearn library [4].

Comparative Analysis of Algorithms

1. Below bar plot shows the comparative analysis of algorithms on Full Dataset. This figure shows that most of the algorithms were unable to predict “Serious” class label.



2. Below bar plot shows algorithm analysis after under sampling the data. This figure shows that every algorithm performs well after under sampling the most frequent class label.



Algorithm Analysis

	PRECISION	RECALL	F1 SCORE	ACCURACY
RANDOM FOREST	93	99	96	92.38
KNN	93	97	95	91
SVM	92	100	96	92.06
LOGISTIC REGRESSION	92	100	96	92
NAIVE BAYES	95	61	74	56.3

Section 4: Discussion & Conclusions

Decisions, difficulties, and discussions:

Our main aim was to predict the severity of the accident when it is “serious” and “fatal.” It was exceedingly difficult to handle this large-sized data. Using HPC we were able to run most of our algorithms. Data is highly imbalanced so even though most of our algorithms were giving > 89% accuracies, it was of no use. It was predicting all the accidents as slight accidents. After checking on all these algorithms, the team even tried dimensionality reduction techniques and, but the results were not improved. Then the team decided to use the under sampled dataset as it was giving better results in predicting severe/fatal accidents. This decision was made on trying out oversampling, under sampling, test, and train data with an equal ratio of classification classes.

Conclusion and Future work:

In conclusion, most of the algorithms are biased towards the most frequent class. However, efficient pre-processing and corresponding imbalanced data techniques should give optimal results. Based on the current known conditions of weather, light, traffic signal, road surface, speed limit etc., accident severity can be classified. But there is no one feature that influences the accident severity.

Future work involves considering region from latitude and longitude and this problem can be turned into regression problem. We can then predict the risk of accidents in the region. If the risk is higher than immediate actions can be taken.

References:

[1] Global Status Report on Road Safety 2019

http://www.who.int/violence_injury_prevention/road_safety_status/2019/en/

[4] Feature selection for machine Learning in Python https://machinelearningmastery.com/feature-selection-machine-learning-python/

[5] https://scikitlearn.org/stable/modules/generated/sklearn.feature_selection.chi2.html#sklearn.feature_selection.chi2