Journal Homepage: -www.journalijar.com
INTERNATIONAL JOURNAL OF ADVANCED RESEARCH (IJAR)

ISSN NO. 2320-5407

Article DOI:10.21474/IJAR01/ 9578
DOI URL: http://dx.doi.org/10.21474/IJAR01/9578

RESEARCH ARTICLE

ACCIDENT SEVERITY CLASSIFICATION USING MACHINE LEARNING.

Kriti Dwivedi, Ajay Singh and Madhav Bharadwaj.

……………………………………………………………………………………………………....

Manuscript Info

……………………….

Manuscript History
Received: 16 June 2019
Final Accepted: 18 July 2019
Published: August 2019

Keywords
Neural Networks, Support Vector Machines, Decision Trees, Classification.

Abstract

………………………………………………………………….

Automobiles are one of the greatest inventions of all times. They have simplified our lives and provided us with tremendous comfort. But there are two sides to every coin. They are one of the major causes of deaths in many countries. Driving while drowsy is a major cause for accidents in India and elsewhere. In India, more than 150,000 people are killed each year in traffic accidents, which is about 400 fatalities a day and far higher than developed countries like the US, which in 2016 logged about 40,000. In fact, as many as one third of fatal car accidents are linked to drowsy driving. In order to reduce these negative effects, it is important that measures be made on the scientific and objective front of the causes of accidents and severity of injuries. This project summarizes the performance of four machine learning paradigms applied to modelling the severity of injury that occurred during traffic accidents.

……………………………………………………………………………………………………....

Introduction:-

Researchers in the automobile industry have tried to design and build safer automobiles, but traffic accidents are unavoidable. Patterns involved in dangerous crashes could be detected if we develop accurate prediction models which are capable of automatic classification of type of injury severity of various traffic accidents. These behavioural and roadway accident patterns can be useful to develop traffic safety control policies. We believe that to obtain the greatest possible accident reduction effects, it is important that measures be based on scientific and objective surveys of the causes of accidents and severity of injuries. This project summarizes the performance of four machine learning paradigms applied to modelling the severity of injury that occurred during traffic accidents. The costs of fatalities and injuries due to traffic accidents have a great impact on the society. In recent years, researchers have paid increasing attention to determining factors that significantly affect severity of driver injuries caused by traffic accidents. There are several approaches that researchers have employed to study this problem. Applying data mining techniques to model traffic accident the data records can help to understand the characteristics of drivers behaviour, roadway condition and weather condition that were causally connected with different injury severity.

Dataset Description

In this project, we used data from the National Automotive Sampling System (NASS) General Estimates System(GES). The GES datasets are intended to be a nationally representative probability samples from the annual estimated 6.4 million accident reports in the United States. The initial dataset for the study contained traffic accident records from 1997 to 2000, a total number of 1,64,302 cases.This dataset has drivers records only and does not include passengers information. The total set includes labels of year, month, region, primary sampling unit, case

Corresponding Author:-Kriti Dwivedi.

number, vehicle number, vehicle make and model, inputs of drivers alcohol usage, restraint system, eject, vehicle body type, vehicle age, vehicle role, initial point of impact, manner of collision, rollover, roadway surface condition, light condition, travel speed, speed limit and the output injury severity. The injury severity has five classes:

- no injury
- possible injury
- non-incapacitating injury
- incapacitating injury
- fatal injury

Our objective was to classify the dataset into these five classes and compare the accuracy obtained.

Proposed Work

Problem Statement

Automobiles have certainly made our lives easier but they are coupled with the risk of accidents. It is one of the Automobiles have certainly made our lives easier but they are coupled with the risk of accidents. It is one of the major causes of deaths in many countries. This project summarizes the performance of four machine learning paradigms applied to modelling the severity of injury that occurred during traffic accidents.

Challenges Faced

The data set from the National Automotive Sampling Sys-tem (NASS) General Estimates System (GES) [2] that we found was available in .ssd format which is an obsolete format and opening it was a big ordeal for us. After trying several techniques we finally managed to retrieve the data to be used in our project. We got help from an online source who converted those files into .csv format for us.

Another problem that we faced was that as not all algorithms that we wanted to implement work on a multiclass dataset, we had to convert the dataset to one versus all to be able to use them.

Experimental Setup And Results Analysis

Preprocessing of Dataset

Round 1: Using Chi Square Test The $\chi 2$ test is a statistical test of independence to deter-mine the dependency of two variables. It shares similari-ties with coefficient of determination, R2 . However, $\chi 2$ test is only applicable to categorical or nominal data while R2 is only applicable to numeric data. From the definition, of $\chi 2$ we can easily deduce the application of chi-square technique in feature selection. Suppose you have a target variable (i.e., the class label) and some other features (feature variables) that describes each sample of the data. Now, we calculate $\chi 2$ statistics between every feature variable and the target variable and observe the existence of a relationship between the variables and the target. If the target variable is independent of the feature variable, we can discard that feature variable. If they are dependent, the feature variable is very important. The $\chi 2$ test indicated that all the variables are significant (p-value 6 0.5).

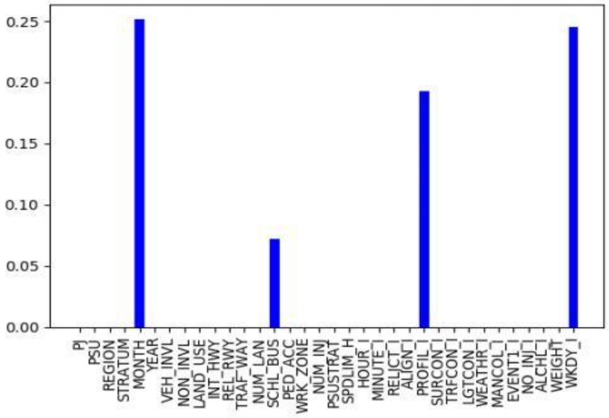


Fig 1:-The results obtained from Chi-squared Test show that the severity of accidents had a strong correlation with factors like the Month, weekday and whether or not the vehicle was a school bus.

Round 2: Using Algorithm

The second thing we tried was to feed the data through feature selection algorithms. The first was a univariate feature selection, which selects the best features based on univariate statistical tests. In particular, we used SelectKBest which is part of the scikit-learn package [7]. We also used Tree-based feature selection, which uses atransform method to reduce the dimensionality of the data.Tree-based estimators can be used to compute feature importances, which in turn can be used to discard irrelevant features.

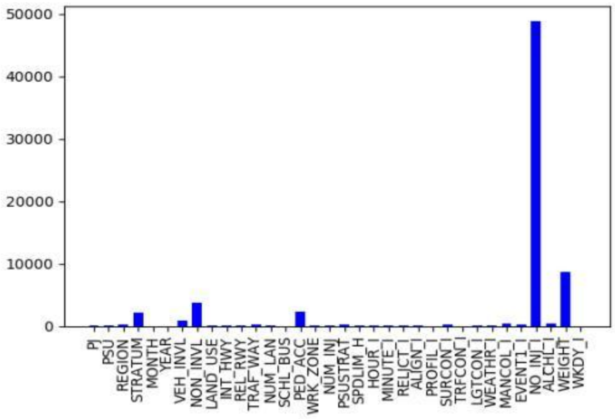


Fig 2:- The results obtained from Select K Best Algorithm showed results different than that of Chi squared Test. Correlation with vehicle weight and number of passengers were observed.

Round 3: Feature Extraction

The data set, as explained above, contains 33 features. We divided them into 3 subsets of 11 features each. Further, we applied SelectKBest Algorithm on each of the 3 sets to calculate the weights of each feature in the subset and multiplied the outcome with the original set of 11 features each. Summation of this yields 3 features which are further used in various models of our project. But we did not obtain any significant improvement in accuracy on using these features instead of the original set of 33 features. This area is still to be explored in detail.

Description of Models

Neural Network

Using the features selected above we, processed them using Neural Networks. We eventually used an implementation provided in the scikit library. A Multilayer Perceptron (MLP) [8] is a feed forward neural network with one or more hidden layers.The algorithm starts with random initializing all the weights {w} and biases {b}.Then output of hidden neurons is calculated as

$$y_i(p) = f[\sum_{i=1}^{n} x_i(p) * w_{ij}(p) + b_j] \qquad (1)$$

where n is the number of inputs of neuron j in hidden layer.Next calculate the actual outputs of neurons in out-put layer. The weight training is to update the weights using the Back-Propagation (BP) learning method with the error function

$$MeanSquaredError = \frac{(ExpectedOutput - PredictedOutput)^2}{2} \qquad (2)$$

The objective of weight training is to change the weight vector w so that the error function is minimized. By minimizing the error function, the actual output is driven closer to the desired output. The network consists of an input layer of source neurons, at least one hidden layer of computational neurons, and an output layer of computational neurons. The input layer accepts input signals and redistributes these signals to all neurons in the hidden layer. The output layer accepts a stimulus pattern from the hidden layer and establishes the output pattern of the entire network. We used two hidden layers with 30 and 50 neurons respectively.Input layer has 33 neurons and the output layer has 5 neurons corresponding to each class for the multi-class classification and 2 neurons for the one-against-all classification. Sigmoid funcion is used as an activation function for the layers and Stochastic Gradient Descent(SGD) is used as an optimization technique in the neural network. We used Principal Compo Analysis (PCA) with 10 principle components to reduce the dimensionality which provided maximum accuracy.

Support Vector Machines

Support Vector Machine (SVM) is based on statistical learning theory. SVMs have been successfully applied to a number of applications ranging from handwriting recog-nition, intrusion detection in computer networks, and text categorization to image classification, breast cancer diag-nosis and prognosis and bioinformatics. SVM involves two key techniques, one is the mathematical program-ming and the other is kernel functions. Here, parameters are found by solving a quadratic programming problem with linear equality and inequality constraints; rather than by solving a nonconvex, unconstrained optimization problem. SVMs are kernel-based learning algorithms in which only a fraction of the training examples are used in the solution (these are called the support vectors), and where the objective of learning is to maximize a margin around the decision surface. The flexibility of kernel func-tions allows the SVM to search a wide variety of hypothe-sis spaces. The basic idea of applying SVMs to pattern classification can be stated briefly as: first map the input vectors into one feature space (possible with a higher di-mension), either linearly or nonlinearly, whichever is re-levant to the selection of the kernel function; then within the feature space, seek an optimized linear division, i.e. construct a hyperplane which separates two classes.

Decision Trees

We implemented the Classification and Regression Trees (CART) algorithm based on decision trees. Decision trees are well-known algorithm for classification problems. The CART model consists of a hierarchy of univariate binary decisions. Each internal node in the tree specifies a binary test on a single variable, branch represents an outcome of the test, each leaf node represent class labels or class dis-tribution. CART operates by choosing the best variable for splitting the data into two groups at the root node, partitioning the data into two disjoint branches in such a way that the class labels in each branch are as homogene-ous as possible, and then splitting is recursively applied to each branch, and so forth. Decision Tree building algo-rithm involves a few simple steps and these are:

1. Take Labelled Input data - with a Target Variable and a list of Independent Variables
2. Best Split: Find Best Split for each of the independent variables using Gini Index
3. Best Variable: Select the Best Variable for the split
4. Split the input data into Left and Right Nodes
5. Continue step 2-4 on each of the nodes until meet stopping criteria
6. Decision Tree Pruning: Steps to prune Decision Tree built

If a dataset T contains examples from n classes, gini index [5], gini (T) is defined as:

$$gini(T) = 1 - \sum_{j=1}^{n} p_j^2 \qquad (3)$$

where pj is the relative frequency of class j in T. If dataset T is split into two subsets T1 and T2 with sizes N1 and N2, the gini index of the split data contains examples from n classes, the gini index gini (T) is defined as

$$ginisplit(T) = \frac{N_1}{N} gini(T_1) + \frac{N_2}{N} gini(T_2) \qquad (4)$$

CART exhaustively searches for uni variate splits. The attribute provides the smallest ginisplit (T) is chosen to split the node. CART recursively expands the tree from a root node, and then gradually prunes back the large tree. The advantage of a decision tree is the extraction of classi-fication rules from trees that is very straightforward. More precisely, a decision tree can represent the know-ledge in the form of if-then rules; one rule is created for each path from the root to a leaf node.

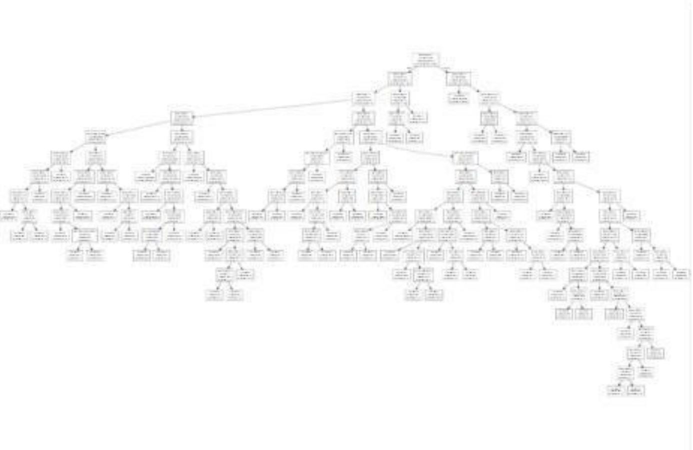


Fig 3:-CART Algorithm was applied to obtain decision trees for various levels of severity.

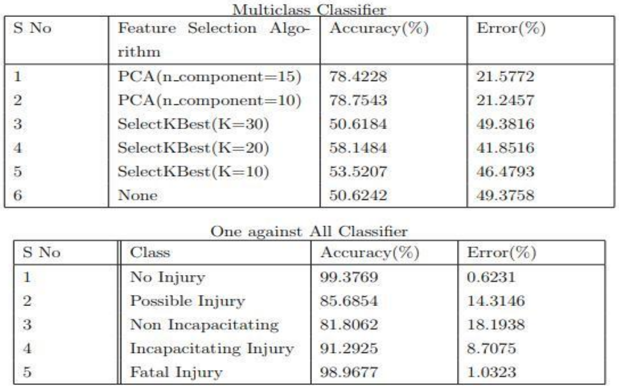
K-nearest Neighbours

K-Nearest Neighbours is one of the most basic yet essen-tial classification algorithms in Machine Learning. It belongs to the supervised learning domain. Euclidean distance between the data points is calculated. We used the scikit learn package to implement this. K-nearest-neighbor classification was developed from the need to perform discriminant analysis when reliable parametric estimates of probability densities are unknown or difficult to determine.It is widely disposable in real-life scenarios since it is non-parametric, meaning, it does not make any underlying assumptions about the distribution of data (as opposed to other algorithms such as GMM, which as-sume a Gaussian distribution of the given data).

Result Comparison:-

Neural Networks

The analysis was done for two kinds of classification, multiclass and one against all. Multiclass essentially tells exactly which class the test input belongs to. One against all on the other hand, tells whether a test input belongs to a certain class or not.



Multiclass Classifier

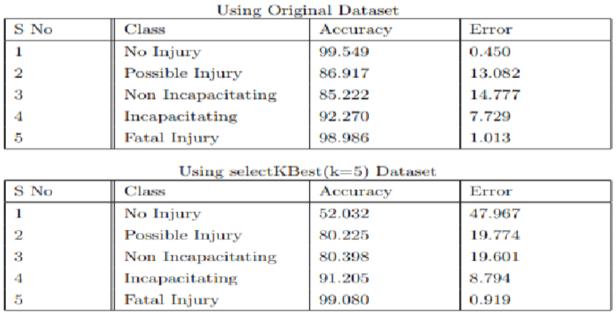
S No	Feature Selection Algorithm	Accuracy(%)	Error(%)
1	PCA(n_component=15)	78.4228	21.5772
2	PCA(n_component=10)	78.7543	21.2457
3	SelectKBest(K=30)	50.6184	49.3816
4	SelectKBest(K=20)	58.1484	41.8516
5	SelectKBest(K=10)	53.5207	46.4793
6	None	50.6242	49.3758

One against All Classifier

S No	Class	Accuracy(%)	Error(%)
1	No Injury	99.3769	0.6231
2	Possible Injury	85.6854	14.3146
3	Non Incapacitating	81.8062	18.1938
4	Incapacitating Injury	91.2925	8.7075
5	Fatal Injury	98.9677	1.0323

Table 1:-Summary of the best results obtained for Neural Networks

Decision Trees



Using Original Dataset

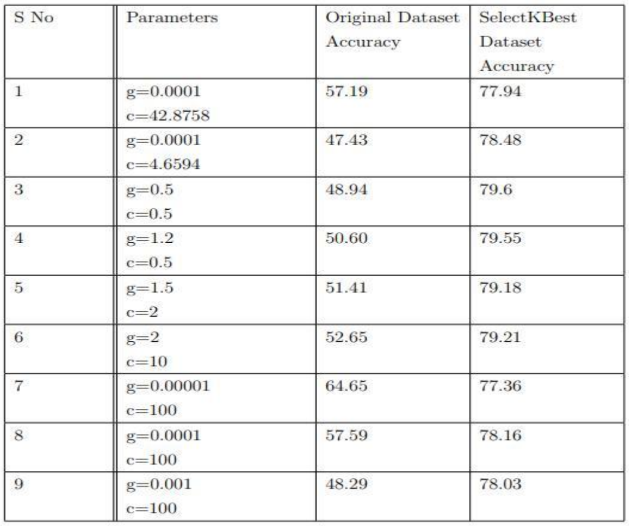
S No	Class	Accuracy	Error
1	No Injury	99.549	0.450
2	Possible Injury	86.917	13.082
3	Non Incapacitating	85.222	14.777
4	Incapacitating	92.270	7.729
5	Fatal Injury	98.986	1.013

Using selectKBest(k=5) Dataset

S No	Class	Accuracy	Error
1	No Injury	52.032	47.967
2	Possible Injury	80.225	19.774
3	Non Incapacitating	80.398	19.601
4	Incapacitating	91.205	8.794
5	Fatal Injury	99.080	0.919

Table 2:-Summary of the best results obtained for Decision Trees

Support Vector Machines



S No	Parameters	Original Dataset Accuracy	SelectKBest Dataset Accuracy

1	g=0.0001
	c=42.8758
		57.19	77.94

2	g=0.0001
	c=4.6594
		47.43	78.48

3	g=0.5
	c=0.5
		48.94	79.6

4	g=1.2
	c=0.5
		50.60	79.55

5	g=1.5
	c=2
		51.41	79.18

6	g=2
	c=10
		52.65	79.21

7	g=0.00001
	c=100
		64.65	77.36

8	g=0.0001
	c=100
		57.59	78.16

9	g=0.001
	c=100
		48.29	78.03

Table 3:-Summary of the best results obtained for SVM

K-nearest Neighbours

1. Accuracy using Original Set of features = 58.49%
2. Accuracy using 3 Extracted features = 42.56%
3. Accuracy using selectKBest(k=5) = 77.89%

Comparison of Algorithms: Multiclass

The following graph represents the accuracy of various Machine Learning Paradigms that we applied to our dataset. It is clear from the chart that Decision Tree with Original Dataset(unpreprocessed) performs the best with an accuracy of 79.95%.

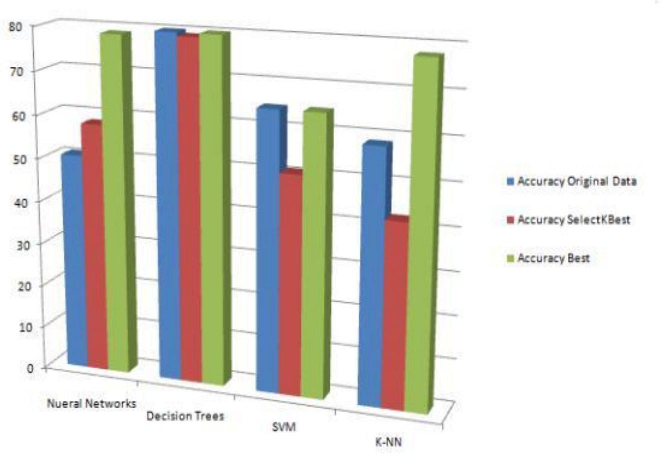


Fig 4:-A comparison of Multiclass Algorithms of all four kinds shows that Decision Trees worked the best for our data set.

Conclusion and Future Work:-

The results clearly show that maximum accuracy is ob-tained by using Decision Tree on the original set of features(33 features). The most weighted features turn out to be:

- Number of Non-Motorists Involved
- Pedestrian/Cyclist Crash Type
- Number Known Injured in Crash
- Case Weight
- Case Stratum

There are several recent data sets that are available. These can be further processed and examined to analyze traffic accidents using the above mentioned machine learning paradigms. This project has a real life application as the need to reduce accidents due to automobiles is a growing concern. Also, we plan on analyzing the time of crash more closely. We might find useful insights about the number of accidents in rush hours versus normal hours. We also plan on applying more Feature Extraction Techniques to the dataset to improve the accuracy. Hybrid Models using both decision trees and neural network can be developed to test for accuracy. Integrating different learning models gives better performance than the individual learning or decision-making models by reducing their individual limitations and exploiting their different mechanisms. In a hierarchical hybrid intelligent system each layer provides some new information to the higher level.

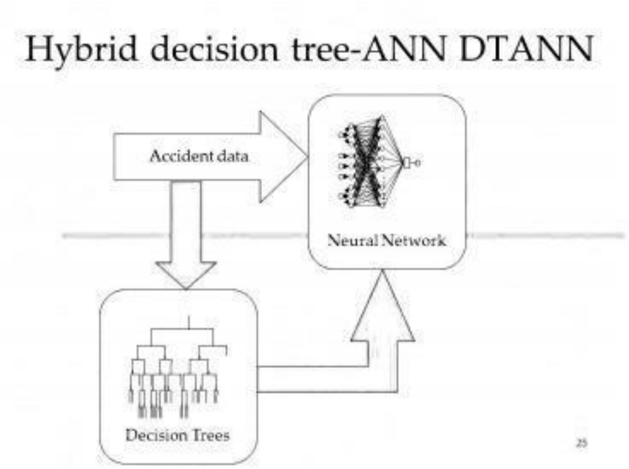


Fig 5: Hybrid models are expected to perform better than individual models in classification.

Acknowledgement:-

We have taken efforts in this project. However, it would not have been possible without the kind support and help of many individuals. We would like to extend our sincere thanks to all of them. We are highly indebted to our supervisor, Dr. Dinesh Singh for his guidance and constant supervision as well as for providing necessary informa-tion regarding the project and also for his support in completing the project. Our thanks and appreciations also go to our colleagues and people who have willingly helped us out with their abilities.

References:-

1. Abraham, A. Intelligent systems: Architectures and perspectives. In Recent advances in intelligent paradigms and applications. Springer, 2003, pp. 1–35.
2. Administration, N. H. T. S., et al. National automotive sampling system (nass) general estimates system (ges) analytical users manual 1988-1999. US Department of Transportation (2000).
3. Al-Khateeb, G. G. Analysis of accident data and evaluation of leading causes for traffic accidents in jordan. Jordan J Civ Eng 4,2 (2010), 76–94.C. J.
4. bin Tariq, T., and Chen, A. Stay alert! the ford challenge.
5. Lerman, R. I., and Yitzhaki, S. A note on the calculation and interpretation of the gini index. Economics Letters 15, 3-4 (1984), 363–368.
6. Olutayo, V., and Eludire, A. Traffic accident analysis using decision trees and neural networks. International Journal of Information Technology and Computer Science (IJITCS) 6, 2(2014), 22.
7. L. Hubert and P. Arabie, ―Comparing Partitions,‖ J. Classification, vol. 2, no. 4, pp. 193-218, Apr. 1985. (Journal or magazine citation)
8. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B.,Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., et al. Scikit-learn: Machine learning in python. Journal of Machine
9. Learning Research 12, Oct (2011), 2825–2830.Taud, H., and Mas, J. Multilayer perceptron (mlp). In Geomatic Approaches for Modeling Land Change Scenarios. Springer, 2018, pp.451–455.