A Comparative Study of Hybridized Neural Networks in Estimating Traffic Accident Severity

Md. Mydul Islam Anik

2016-1-60-081

Wasim Akram

2016-1-60-091

Md. Ashikuzzaman

2016-1-60-094

A thesis submitted in partial fulfillment of requirements for the degree of Bachelor of Science and Engineering

Department of Computer Science and Engineering

East West University

Dhaka-1212, Bangladesh

December, 2019

Declaration

We, hereby, declare that the work presented in this thesis is the outcome of the investigation performed by us under the supervision of **Dr. Md Sawkat Ali**, Assistant Professor, Department of Computer Science and Engineering, East West University. We also declare that no part of this Thesis/project has been or is being submitted elsewhere for the award of any degree or diploma.

....

Dr. Md Sawkat Ali
Assistant Professor
Department of Computer Science & Engineering
Supervisor

....

Md. Mydul Islam Anik
2016-1-60-081
Department of Computer Science and Engineering
East West University

....

Wasim Akram
2016-1-60-091
Department of Computer Science and Engineering
East West University

....

Md. Ashikuzzaman
2016-1-60-094
Department of Computer Science and Engineering
East West University

Letter of Acceptance

This thesis report titled “A Comparative Study of Hybridized Neural Networks in Estimating Traffic Accident Severity” submitted by Md. Mydul Islam Anik (ID: 2016-1-60-081), Wasim Akram (ID: 2016-1-60-091) and Md. Ashikuzzaman (ID: 2016-1-60-094) to the Department of Computer Science and Engineering, East West University, Bangladesh is accepted as satisfactory for the partial fulfilment of requirements for the Award of the Degree of Bachelor of Science in Computer Science and Engineering on December,2019.

Supervisor

…………………………..

(Dr. Md Sawkat Ali)

Assistant Professor

Department of Computer Science and Engineering

East West University, Bangladesh

Chairperson

…………………………..

(Dr. Taskeed Jabid)

Associate Professor

Department of Computer Science and Engineering

East West University, Bangladesh

Acknowledgement

As it is true for everyone, we have also arrived at this point of achieving a goal in our life through various interactions with and help from other people. However, written words are often elusive and harbor diverse interpretations even in one's mother language. Therefore, we would not like to make efforts to find best words to express my thankfulness other than simply listing those people who have contributed to this thesis itself in an essential way. This work was carried out in the Department of Computer Science and Engineering at East West University, Bangladesh.

First of all, we would like to express our deepest gratitude to the almighty for His blessings on us. Next, our special thanks go to our supervisor, "Dr. Md Sawkat Ali" who gave us this opportunity, initiated us into the field of Machine Learning, and without whom this work would not have been possible. His encouragements, visionaries and thoughtful comments and suggestions, unforgettable support at every stage of our B.Sc. study were simply appreciating and essential. His ability to muddle us enough to finally answer our own question correctly is something valuable what we have learned and we would try to emulate, if ever we get the opportunity.

We would like to thank Sir for his overall support and his valuable suggestion during our working session. Last but not the least, we would like to thank our parents for their unending support, encouragement and prayer.

Acknowledgement

There are numerous other people too who have shown me their constant support and friendship in various ways, directly or indirectly related to our academic life. We will remember them in our heart and hope to find a more appropriate place to acknowledge them in the future.

Md. Mydul Islam Anik
2016-1-60-081
December, 2019

Wasim Akram
2016-1-60-091
December, 2019

Md. Ashikuzzaman
2016-1-60-094
December, 2019

Table of Contents

Abstract

The increasing number of populations causing increase of vehicles which leads to traffic accident. As transportation system expands, it needs to be monitored to assure safety to citizen. Cities are trying to adopt technological advancement in order to minimize traffic accident. Traffic accidents have become one of the largest national health issues and many factors like weather condition, road condition, light condition, etcetera is related to it. In the current paper, several hybridize machine learning models are used on dataset of city Leeds, UK to estimate traffic accident severity. Hybridize Machine learning models are Artificial Neural Network (ANN) with Gradient decent, Principle Component Analysis (PCA) with ANN, Genetic Algorithm with ANN, Particle Swarm Optimization with ANN. These models are also compared with other machine learning models such as Support Vector Machine (SVM), Naïve Bayes, Nearest Centroid, Logistic Regression, K Nearest Neighbor Classification and Random Forest. Comparison was done considering performance evaluation of each model's accuracy result. Genetic Algorithm with ANN showed promising result of 86.63% accuracy which is the highest score of all model results. Whereas, Nearest Centroid Method gave 55% of accuracy resulting lowest of all. The Results and findings obtained in this study are significant which can provide invaluable information on reducing traffic accident.

Chapter 1

Introduction

As world is going through modernization era, cities are being developed to be “smarter”. Artificial Intelligence is indispensable component of smart cities management. Smart cities are results of evolution in information and communication technologies as a form of digital infrastructure with the physical elements [1]. As world population grows and vehicles are also manufactured for fulfilling the needs. The number of road accident increases, causing severe casualties. Many factors are involved in traffic crashes such as road condition, weather condition, lighting condition and many more. Traditional measures to reduce road accident requires improved geometric design, congestion management strategies and better driver education and enforcement, which will take great period of time and it is not feasible[2]. In order to work with all the features of traffic accident machine learning models are best suited. The model of crash prediction (safety performance function) is one of the most important techniques for investigating the relationship between crash occurrence and risk factors associated with various traffic entities [3]. Various model can be used to train and test upon a dataset to see prediction value in order to evaluate performance of that model. In this paper, significant results were found towards measuring traffic accident severity.

1.1 Motivation

Traffic accident are primary concern due to massive casualties, fatalities and economic losses every year. Many accident severity prediction models are critical to enhancing the safety performance of road traffic system. Hybrid artificial neural network models have effective results towards accident severity prediction. Therefore, implementing many hybrid ANN will result more accurate model to resolve traffic accident. If prediction accuracy is high of a model then it will help to take action towards certain accident. This will be effective and feasible solution towards predicting traffic accident severity.

1.2 Objective

Our main objectives are as follows:

1. Developing several Hybrid artificial neural network to predict traffic accident severity

2. Providing comparison study among Hybrid artificial neural network models and some other machine learning models.

Chapter 2

Related Works

2.1.1 ANN-Gradient Decent Model

An artificial neural network is a computer program that can detect patterns in a given data collection and create a model for that data and it is a pool of specific processing units that transmit signals through a large number of weighted connections to each other. ANN is basically some nodes or units connected together. These nodes are known as artificial neurons. These nodes are connected like biological brain and each nodes or neurons pass information or data to each other. In Artificial Neural Network (ANN) optimizing weights is the ultimate goal. Getting the optimal output is the main purpose of optimization. But in machine learning by optimizing we learn what areas of our data we want to improve [4]. Gradient Decent algorithm is the most used optimizing technique in ANN. ANN normally uses back propagation to determine the result. ANN is used in many areas and traffic accident severity is one of them. ANN in road accident severity is not new. In a study of severity prediction using ANN got an accuracy of 74.6%. That study showed that ANN out performed order probit model [5]. Another study showed ANN accuracy of 61.4% for traffic accident severity prediction. Though Random Forest (80.6%) showed better accuracy in that case [6]. In another research study showed ANN accuracy of 80% for prediction of red light running [22].

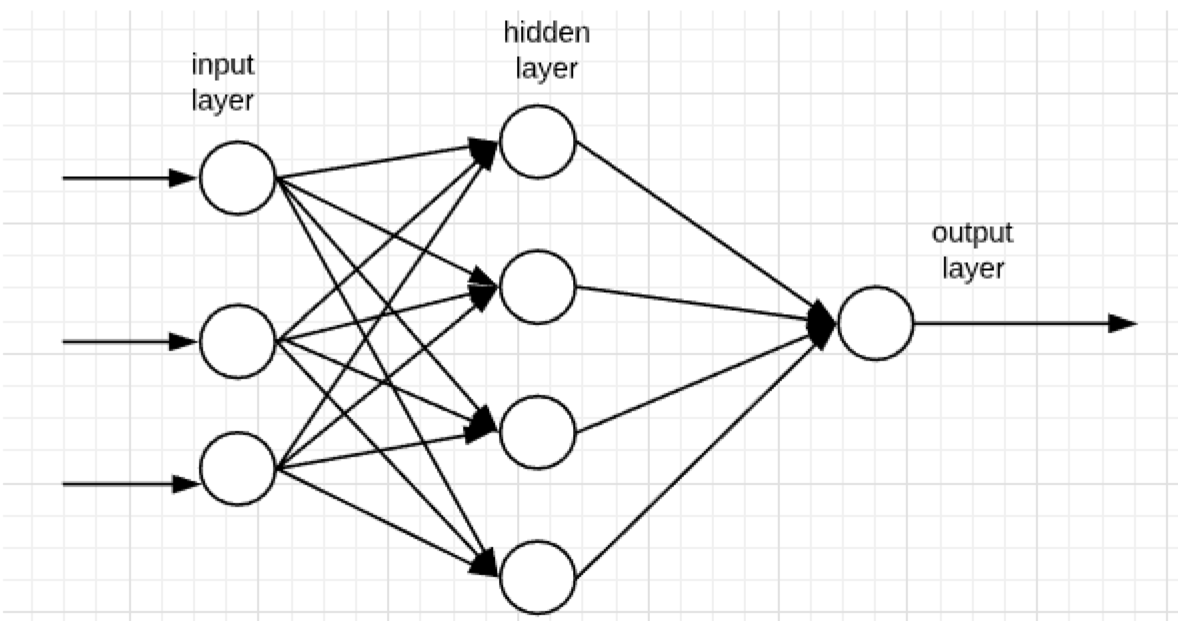

Figure 1: ANN- Gradient Descent Model Diagram

2.1.2 PCA-ANN Model

Principle Component Analysis is a dimensionality reduction method used to reduce the attributes of a dataset that splits four or more features into two or three dimensions, the reduction of attributes can provide a clear visualization of a dataset as well as the correlation between attributes and eliminate features, which are more important than the other features of the dataset. However, the major drawback of this PCA is that it may reduce the accuracy of a machine-learning model. PCA is usually calculated by using the covariance digestion of a data or single value digestion of a data matrix, after a min max normalization or scaling of a standard scalar, both are used to reduce the feature limit. The PCA is used to provide a recommendation system for a website as well as visualize the supervised and unsupervised dataset in machine learning and data mining. At first, we have to use standardized data then we have to calculate covariance matrix and finally we find eigenvalue and eigenvector of covariance matrix. PCA has been used before to optimize neural network before. A study shows PCA-ANN reduced features and gained an accuracy of 91.97% [7]. Another study showed an accuracy of 90% in a face recognition method [8]. Principle Component Analysis with Artificial Neural Network (PCA-ANN) model, which is the first artificial neural network model used in dimension reduction analysis [11]. As the number of input variables increases, the neural network scale will become faster, which called the dimension disaster. PCA is an effectual model to resolve this problem [9].

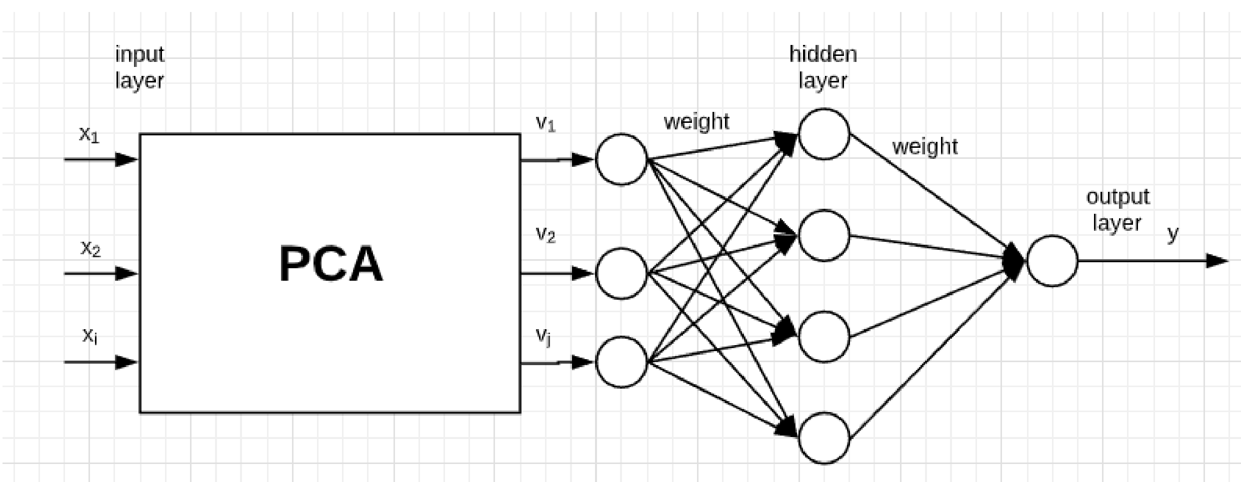

Figure 2: PCA-ANN Model Diagram

2.1.3 GA-ANN Model

The Evolutionary Algorithm (EA) is a type of global neural network adaptation techniques for training or design [9]. The source related algorithm (GA) is a special kind of EA, which is stochastic algorithm when an objective function is called a fitness function is optimized [9]. A starting ANN model was first created by the GA-ANN model; subsequently, genetic algorithm was used to optimize the improved artificial neural network model, where weight and bias were optimized [12]. In general, there is usually a GA-ANN model has several steps; like, first of all to initialize neural networks architecture then encoding, then training the net with initial weight then to calculate the error under each code chain and determine the degree of fitness for each unity function then to select the individual with the largest fitness parent then to selection the crossover and mutation population reproduction when people have chosen to form a new population finally to train the neural network until the optimization solution is found [5].GA-ANN out performed many machine learning models and predicted cancer diagnosis with an accuracy of 97.00% [21]. Another study showed an accuracy of 89.96% in an air quality predicting method [20].

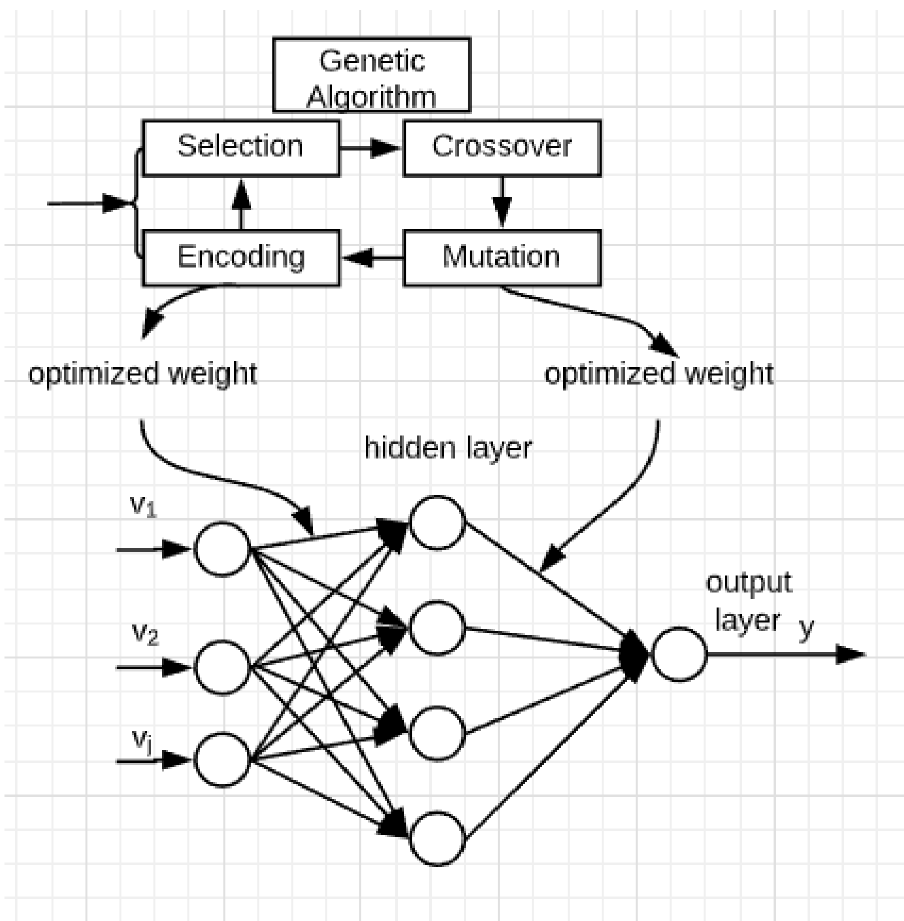

Figure 3: GA-ANN Model Diagram

2.1.4 PSO-ANN Model

Particle Swarm Optimization (PSO) is the technique, which was developed by Kennedy and Eberhart in 1995 [5]. It uses a simple mechanism that impacts swarm behavior in birds flocking and fish schooling to guide the particles to search for global optimal solutions. PSO is proved to be a very efficient optimization algorithm by searching for completely high dimensional problem space [8]. PSO problem does not used because the gradient optimized, so it does not require optimization methods. PSO can be useful for random problem optimization. PSO is initialized with a group of random particles and then searches for optima by updating generations. In every iteration, each particle is updated by following two best values. Like, the first one is the best solution it has achieved so far. This value is a personal best and called pbest. Another best value to be tracked with the particle swarm optimizer is the best value yet achieved by any particle in the population. This value is a global best and called gbest. PSO-ANN model is an optimization algorithm that combines the particle swarm optimization with the artificial neural network. The Particle Swarm Optimization (PSO) algorithm is a global algorithm, which is the powerful ability to find optimistic results around the world [9]. On the other hand, the Artificial Neural Network (ANN) algorithm, which is a strong ability to find local optimistic results but the ability to look for global optimistic results is weak. By combining particle swarm optimization with artificial neural network, the basic idea of this hybrid algorithm is to start searching for the best. In this model, there are many several steps; at first initialize, artificial neural network and particle swarm optimization then optimize ANN with PSO then train and test PSO-ANN prediction model finally analysis the simulation results. A study shows PSO-ANN outperformed SVM with an accuracy of 99.7% [14].

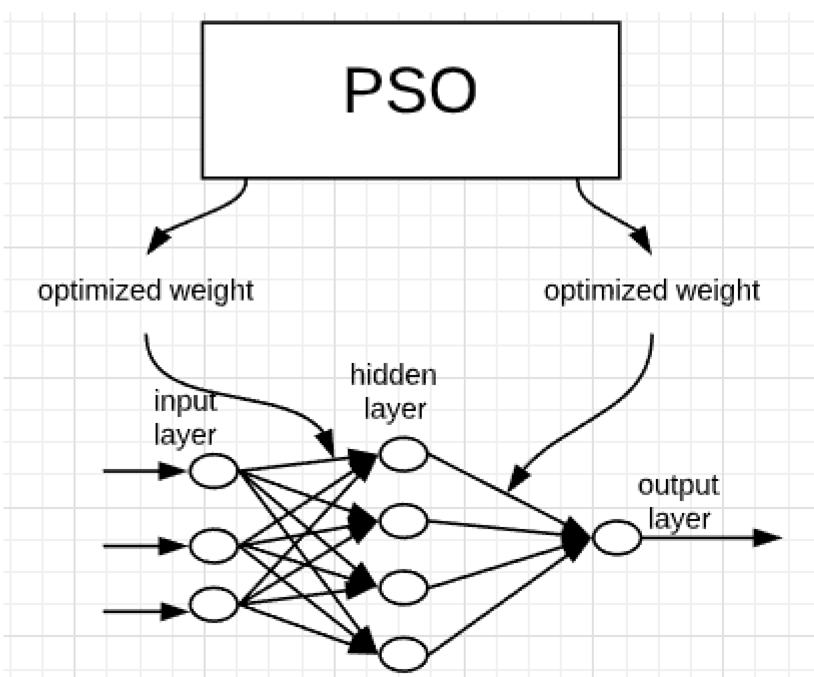

Figure 4: PSO-ANN Model Diagram

2.1.5 Support Vector Machine (SVM)

Support vector machine is a popular machine-learning algorithm for classification and regression algorithm, which was proposed by Vapnik in1960 [10]. Support vector machine is a supervised learning method that looks at data and sorts it into one of the two categories. It is specially a hyperplane that is used as the boundary of different class decisions [11]. The individual instances of these classes are called support vector [11]. The support vector machine has many advantages from other techniques [10]. The advantages of SVM are high dimensional input space, sparse document vectors, regularization parameter. SVM can get good classification results without much training data. SVM can detect optimal classification surfaces at these high dimensional features. While support vector machine is a linear classifier technique, it can model nonlinear interaction using a kernel function by mapping the original input space to a high dimensional feature space [11].

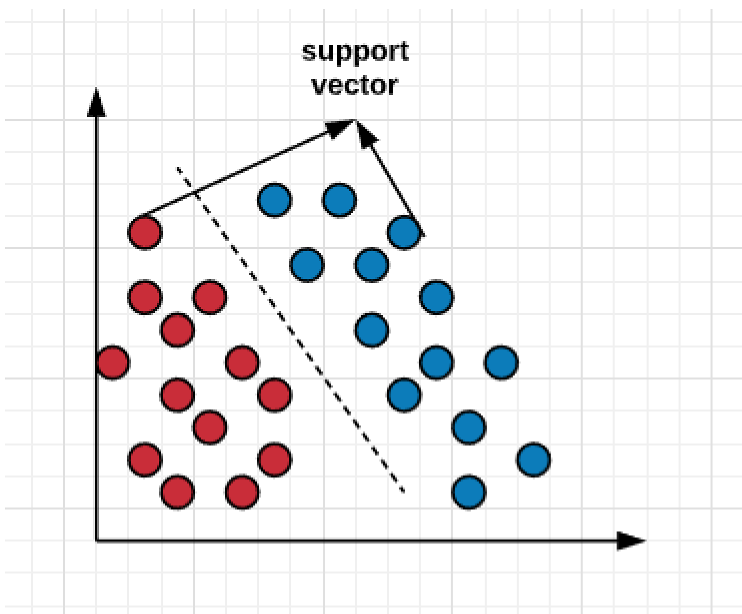

Figure 5: SVM Diagram

2.1.6 Naïve Bayes (NB)

Naïve Bayes is a statistical method for classification and it is a supervised learning method. The Bayes Theorem proposed by Thomas Bayes. The naïve Bayes model is based on the independent concept and the Bayes principle of features [12]. It is usually characterized by *P* (*A* | *B*) which *A* and *B* events [13]. The Probability of *A* is given and *B* is shown in equation (1):

$$P\ (A\ |B) = \frac{P\ (B\ |A)\ P\ (A)}{P(B)} - - - - (1)$$

Using Bayes theorem, we can find the probability of *A* happening, given that *B* has occurred. Here *B* is the evidence and *A* is the hypothesis.
Therefore for the classification of data, the Naïve Bayes classification [13] can be applied as equation (2):

$$P\ (class \mid feature) = \frac{P\ (feature \mid class)\ P\ (class)}{P(feature)} - - - - (2)$$

Where $P\ (class \mid feature)$ is the probability of class when feature occurs, $P(feature)$ is the probability of feature occurrence and $P\ (class)$ is the probability of class occurrence [13].

If each has multiple features, it can be rewritten as equation (3):

$$P\ (class \mid f_1, f_{2,} \dots f_n) = \frac{P\ \left(f_1, f_{2,} \dots f_n \mid class\right)\ P\ (class)}{P\left(f_1, f_{2,} \dots f_n\right)} - - - - (3)$$

The rule of multiplication and total probability can be extended to the probability of the theoretical conditions. Thus, equation (4) extends:

$$P\ \left(f_1, f_{2,} \dots f_n \mid class\right) = \prod_{i=1}^{n} P\ (f_i \mid class) - - - - (4)$$

Finally, there may be ideas for classifying naïve Bayes considered as applying probability $P\ \left(f_1, f_{2,} \dots f_n \mid class\right)$, which depends on the data, was learned when c_j is a member of classes. Considers the classification as high probability equation (5):

$$Classification\ (data) = \arg max_{c_j € c}\ P(c_j) \prod_{i=1}^{n} P\ (f_i \mid c_j) - - - - (5)$$

2.1.7 Logistic Regression (LR)

Logistic Regression is a statistical analyzing technique in which dependency between two variables have been calculated. The main goal of Logistic Regression is to estimate the values of parameter co-efficient. Linear regression has also the same goal but the key difference between linear regression and logistic regression is whether linear regression use linear function to transform the output, Logistic Regression uses logistic function in this case, which is also called log sigmoid function. Logistic regression used in machine learning both classification and regression and it is very efficient over deep neural network because of its efficiency and requirements of less computational resources. Logistic Regression has low variance, which reduces the probability of over fitting. Image segmentation and categorization, geographic image processing, handwriting recognition and healthcare are most popular applications of logistic regression in machine learning.

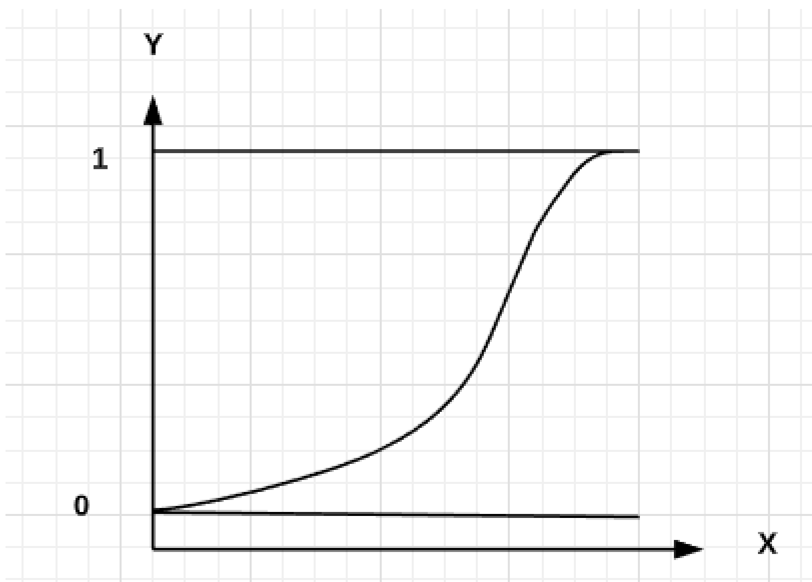

Figure 6: Logistic Regression Diagram

2.1.8 K Nearest Neighbor (KNN)

K nearest neighbor is a powerful classification used in pattern recognition. K nearest neighbors stores all available cases and classifies new cases based on a distance function. It is a non-parametric lazy learning technique. The object is assigned to the most common class in its nearest neighbors. It can use our prior knowledge about which features are more important. Its several kinds of strengths such as KNN is very simple and intuitive and it can be applied to the data from any distribution and it is an excellent classification if the number of samples is large enough. Also, it has many disadvantages such as KNN which is choosing k may be tricky and it has need large number of samples for accuracy.

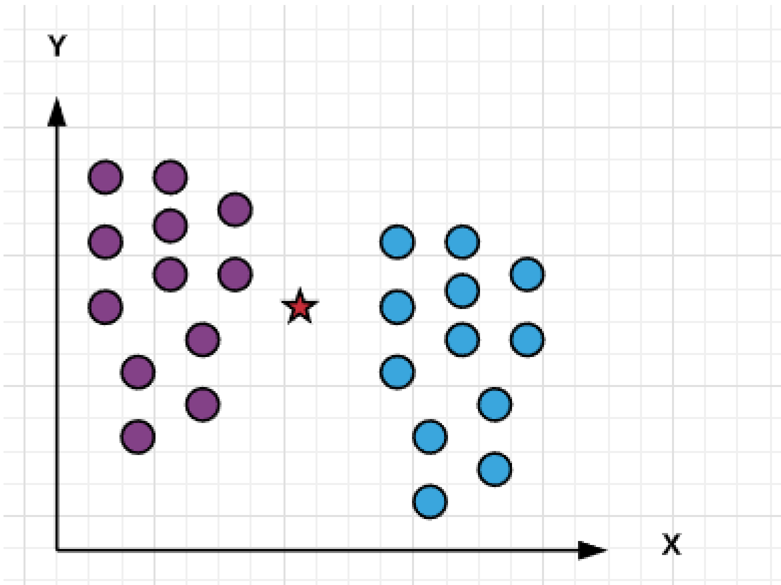

Figure 7: KNN Diagram

2.1.9 Random Forest (RF)

Random Forest classifier is an ensemble classifier using many decision tree models. It can be used for classification or Regression. Random forest classifiers accuracy and variable importance information is provided with the results. Random forest classifier has several applications, such as Remote Sensing which is used in ETM devices to acquire images of the earth’s surface and its accuracy is higher and training time is less and another one is Object Detection which has multiclass object detection is done using random forest algorithms and provides better detection in complicated environments. Random forest has no overfitting and its accuracy is high and it can maintain accuracy when a large proportion of data is missing.

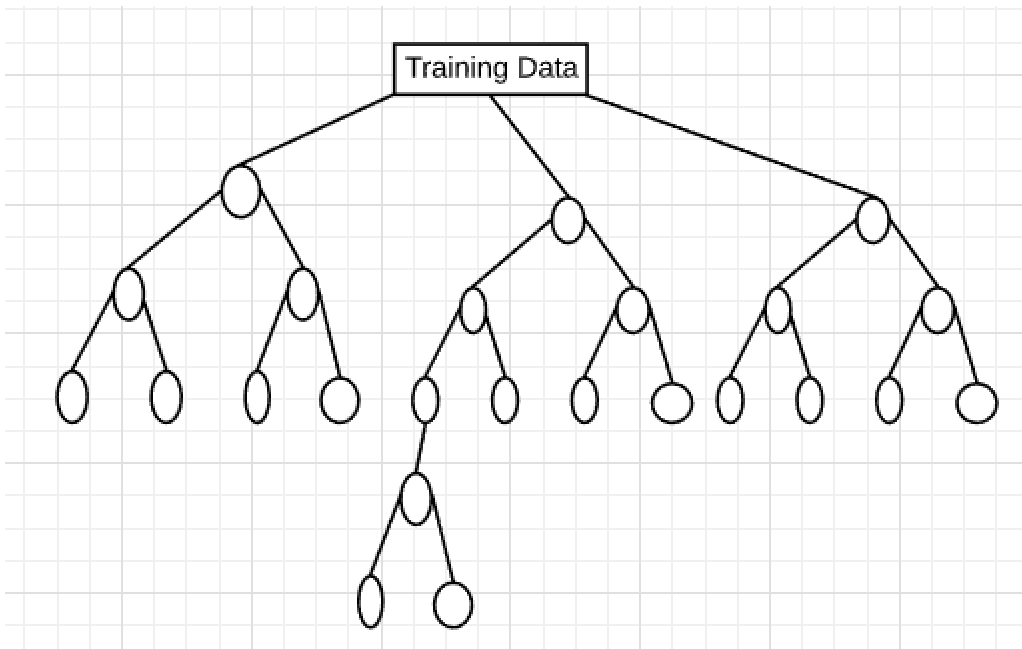

Figure 8: Random Forest Diagram

2.1.10 Nearest Centroid (NC)

Nearest centroid is represented by its centroid, the test specimens are classified into the class with the nearest centroid. The nearest centroid neighbor rule is one of the most effective algorithms for classifying patterns. The nearest centroid classifier to machine learning is a classification model that assigns observations to the class labels of training samples for observation, which is closest to the centroid monitoring.

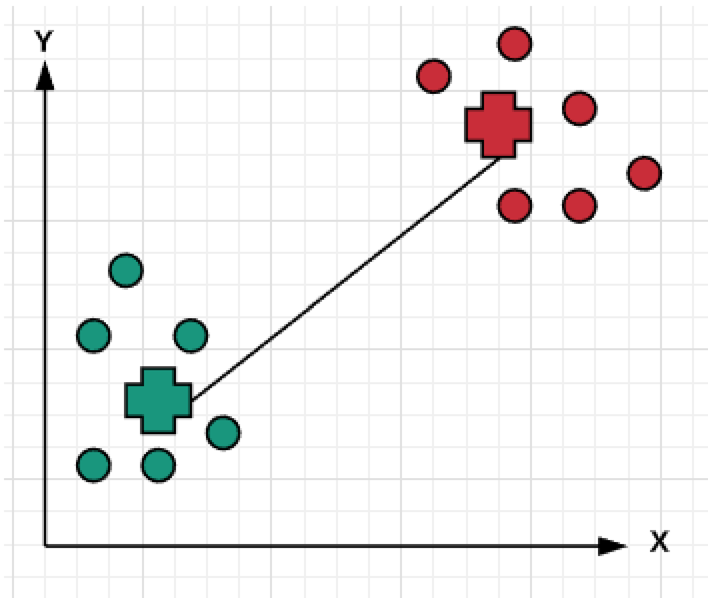

Figure 9: Nearest Centroid Diagram

Chapter 3

Dataset Overview

3.1 Dataset Description

The dataset used in this paper is the Road Accident and Safety data that is collected from (https://data.gov.uk/dataset/6efe5505-941f-45bf-b576-4c1e09b579a1/road-traffic-accidents/datafile/89bf272f-c09f-4216-af7d-feb78ea94cbb/preview) which was published by Department for Transport of the United Kingdom in the year 2018. Dataset's data is of year 2017. The dataset is related to environmental factors containing 2203 traffic accident record with 15 features. One of the feature is class label (Casualty Severity). Reference number and Accident date features were eliminated due to they are not that important for processing data. So total 13 features including class label are being worked on. There is no missing value.

Table 3.1- Dataset Description

Feature Name	Feature Description and Values
Grid Ref: Easting	Grid numbers on the east-west (horizontal) axis are called Eastings. Values are integer type.
Grid Ref: Northing	Grid numbers on the north-south (vertical) axis are called Northings. Values are integer type.
Number of Vehicles	Number of vehicles involved in an accident. Numbers are 1, 2, 3, 5, 4, 6, 7.
Time (24hr)	Time of accident occurred. Values are integer type.
1st Road Class & No	Data type is object. Values sample 'A643', 'A61', 'A653', 'U etc
Road Surface	1. Dry 2. Wet/Damp 3. Frost/Ice 4. Snow
Lighting Conditions	1. Daylight: Street lights present 2. Darkness: Street lights present and lit 3. Darkness: No street lighting 4. Darkness: Street lighting unknown 5. Darkness: Street lights present but unlit 6. Darkness: Street lights present and lit and lit
Weather Conditions	1. Other 2. Fine without high winds

	3. Raining without high winds 4. Fine with high winds 5. Fog or mist (if hazard) 6. Raining with high winds 7. Snowing with high winds 8. Snowing without high winds
Type of Vehicle	1. Car 2. Pedal cycle 3. Motorcycle 12Motorcycle over 500cccc to Motorcycle over 500cc00cc 4. Motorcycle Motorcycle over 500cc0cc and under 5. Taxi/Private hire car 6. Car0 7. Pedal cyclePedal cycle 8. Motorcycle Motorcycle over 500cc0cc to 12Motorcycle over 500cccc 9. Pedal cycleCar 10. Motorcycle Motorcycle over 500cc0cc and underPedal cycle 11. Motorcycle Motorcycle over 500cc0cc and under0 12. Motorcycle over 500cc 13. Car7 14. Pedal cycle0 15. CarTaxi/Private hire car 16. Motorcycle Motorcycle over 500cc0cc and underMotorcycle Motorcycle over 500cc0cc and under
Casualty Class	1. Pedestrian 2. Driver or rider 3. Vehicle or pillion passenger
Casualty Severity	1. Serious 2. Slight 3. Fatal
Sex of Casualty	1. Female 2. Male
Age of Casualty	Unique Ages sample 61, 36, 32, 30, 1,…

Table 3.2- Class label Description

Casualty Severity Label	Code	Number of Instances
Fatal	0	15
Serious	1	309
Slight	2	1879

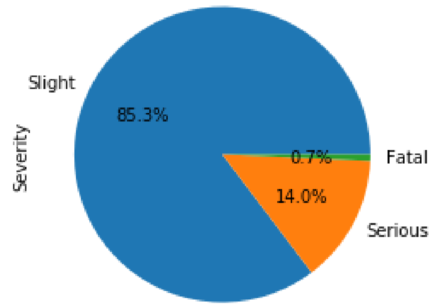

Figure 3.1: Class label is shown in pie chart with percentage value.

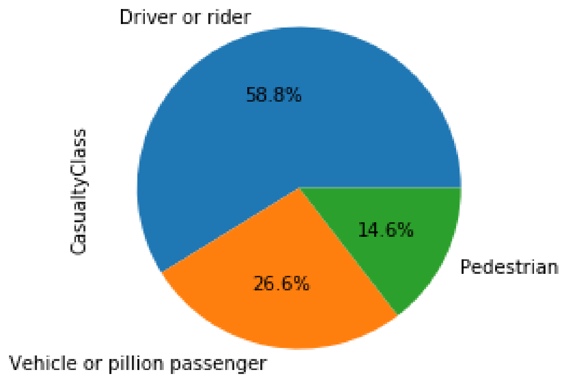

Figure 3.2: Casualty Class is shown in pie chart

According to the figure 3.2, Driver or rider takes more casualties than others.

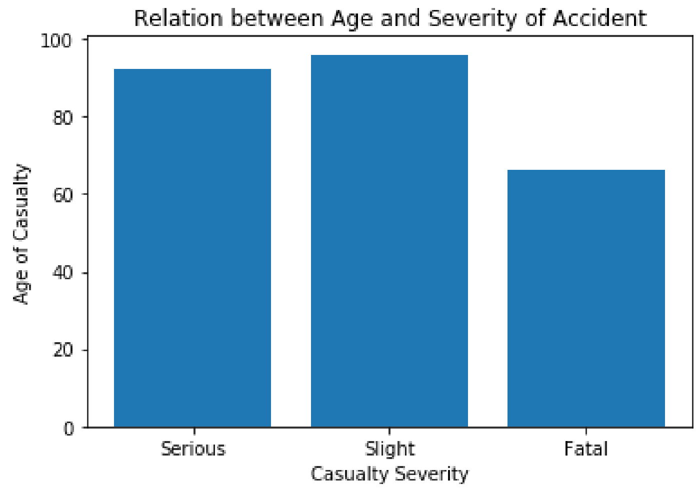

Figure 3.3: Bar chart of relation between Age and Severity of accident

According to Figure 3.3, Age of under 80 people had fatal injuries. Severity level slight occurred to most of all aged people.

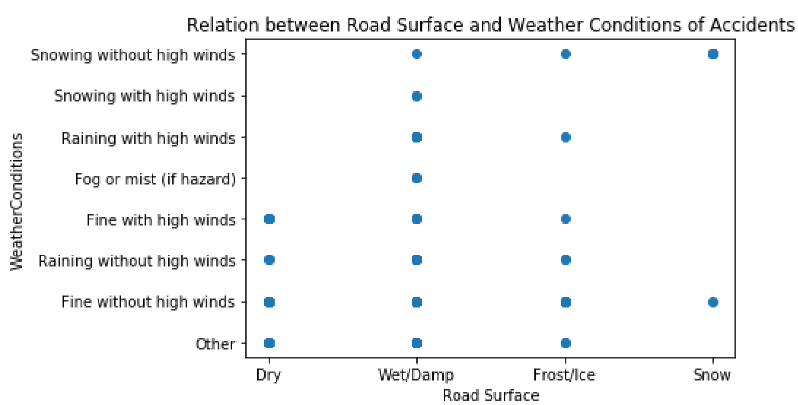

Figure 3.4: Scatter diagram of relation between Road Surface and Weather Condition of Accidents

According to figure 3.4, all the weather condition is reason of wet/dry road surface.

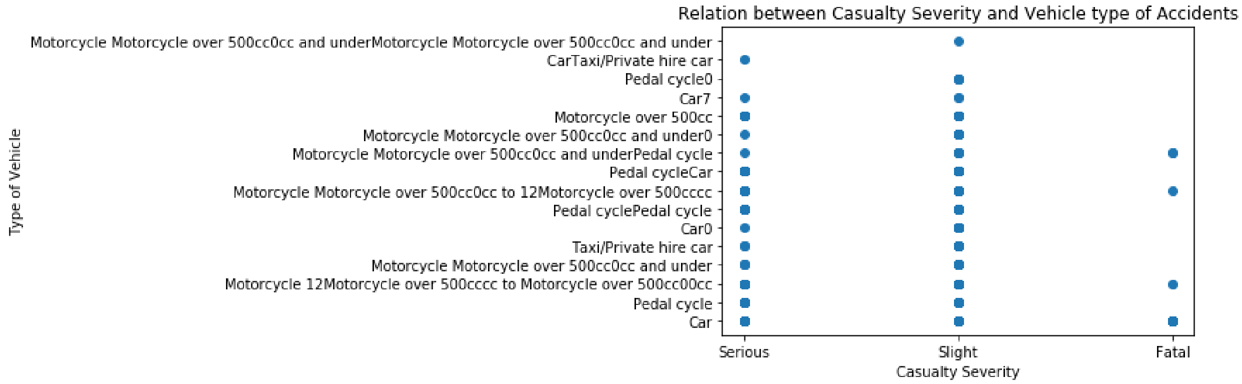

Figure 3.5: Scatter diagram of relation between Casualty Severity and Vehicle type

According to Figure 3.5, motorcycle above 500cc and cars deals with all sort of casualty severity.

3.2 Data Preprocessing

Many Data in Dataset are Categorical data. Many machine learning algorithms cannot operate on label data directly. So, input and output variables must be numeric. In order to work our dataset had to be encoded so that machine learning models can do their tasks efficiently and correctly. One hot encoding was implemented in our dataset. One hot encoding converted categorical data to integer data. After encoding the dataset, we used it in various machine learning algorithms.

Chapter 4
Methodology

4.1.1 ANN-Gradient Descent:

After selecting dataset, we preprocessed it with one hot encoding. Replaced all missing values with 0. Now it was time to select number of hidden layers, learning rate and number of iterations. We set hidden layers number to 5, learning rate to 0.1 and number of iterations to 400. We used cross validation to set train and test data. So, we set number of cross validations to 4. Number of input dimension was 12 and number of output dimension was 3 as there was 12 attributes and 3 different classes (Fatal, Serious, Slight). We then set weights to random values. Next, we used sigmoid function for the purpose of forward propagation. After forward propagation it was time for the backward propagation. We first determined the error rate and then to reduce that error gradient descent was applied. The process continued till the number of iterations was reached. Then based on the result prediction was made.

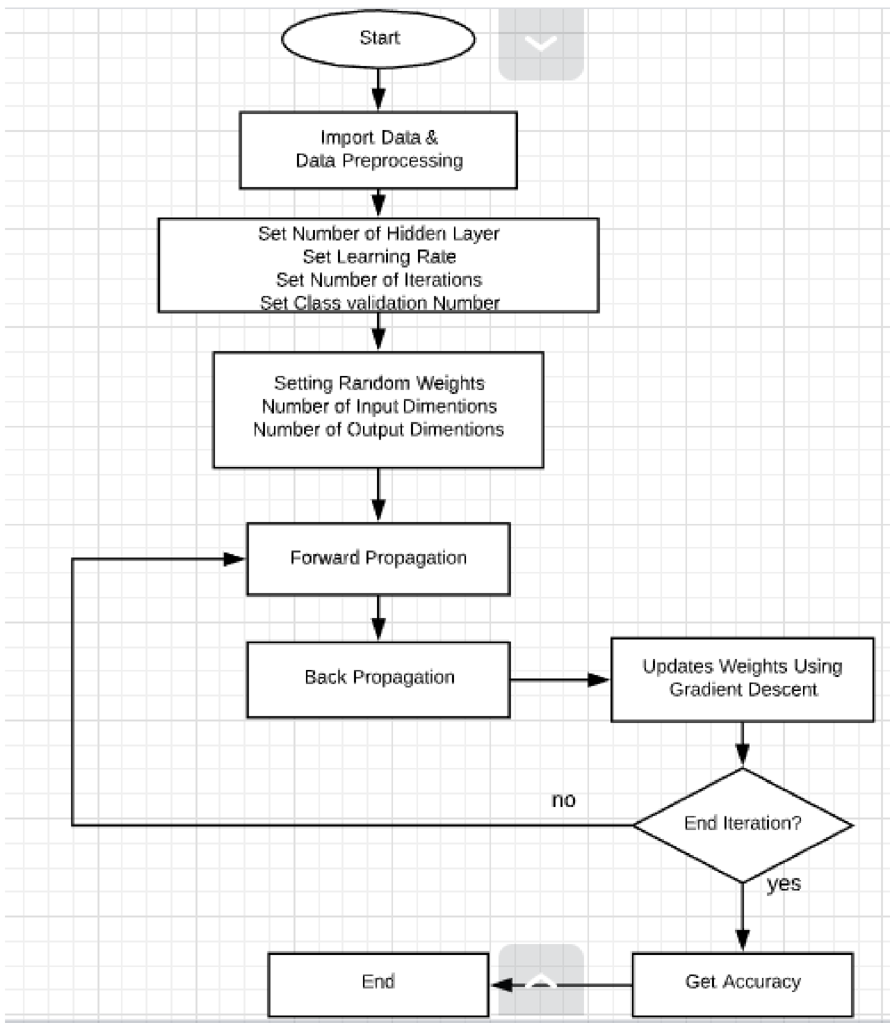

Figure 4.1: Flowchart of ANN-Gradient Descent

4.1.2 PCA-ANN:

PCA is normally applied to reduce the dimension when there are more attributes. It is used to reduce execution time and complexity of attributes. We applied PCA and converted our 12 input dimension to 2 input dimension. Next we saved the data to a new csv file. On that dataset we applied Artificial Neural Network described above.

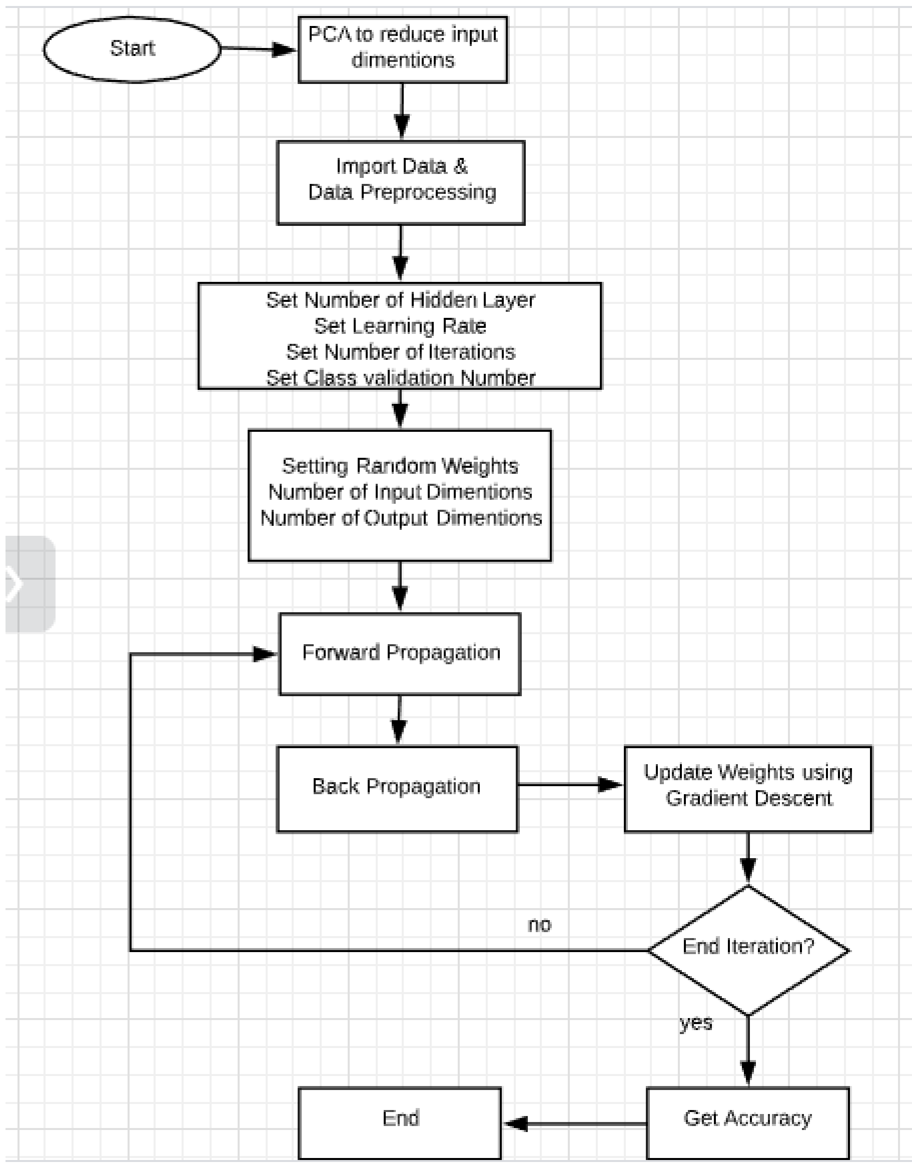

Figure 4.2: Flowchart of PCA-ANN

4.1.3 GA-ANN:

To improve accuracy, we used Genetic Algorithm instead of backpropagation. We used 5 hidden layers, set random weights and biases and number of iteration to 400. We used one hot encoding technique here too. The main deference was after the forward propagation it was the genetic algorithm what we used to update the weights. It continued till the number of iteration reached. And finally calculated the accuracy of prediction.

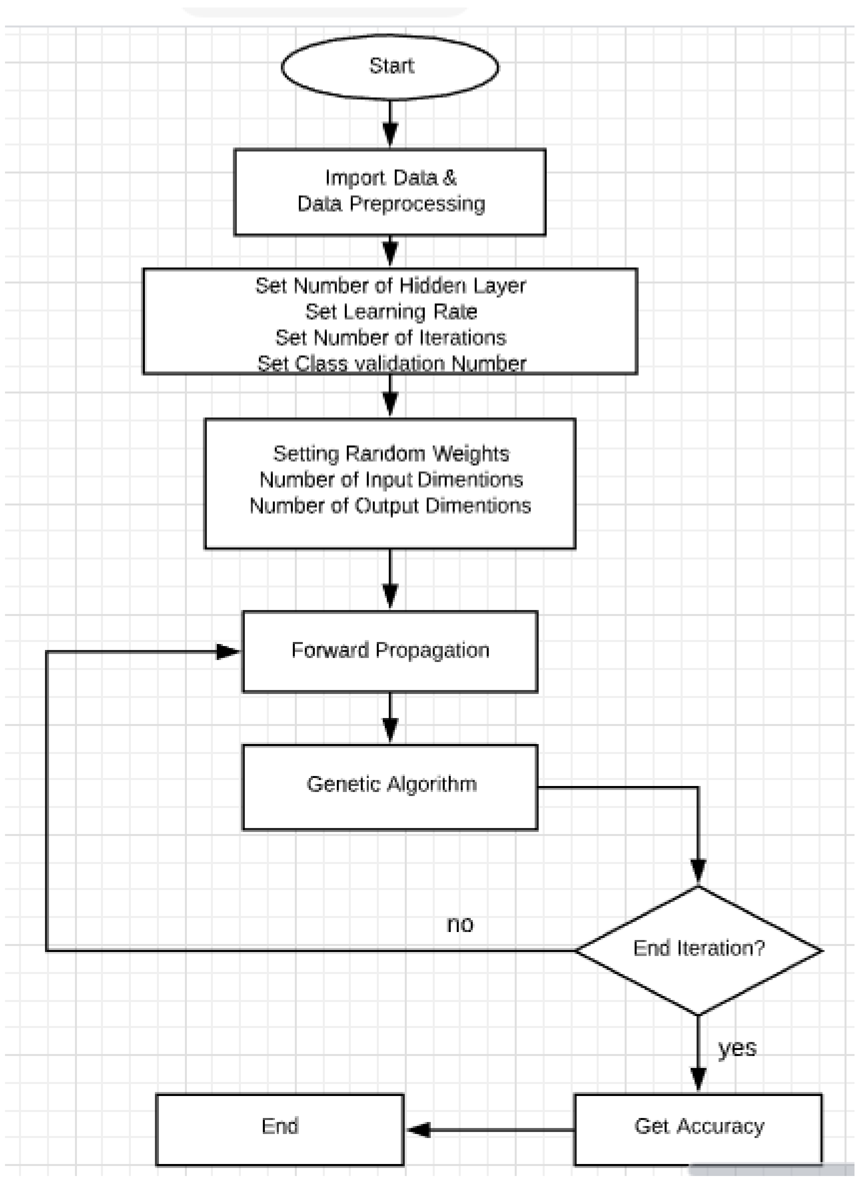

Figure 4.3: Flowchart of GA-ANN

4.1.4 PSO-ANN:

PSO is an optimizing technique that we used to update the weights of Neural Network instead of Gradient Descent Optimizer. We set the hidden layers to 5 and 12 input dimensions and 3 output dimensions. First we used forward propagation. Then we used PSO, the global best value to update the weights and used the forward propagation again. Finally we got the accuracy of prediction. To determine the dimension of global best value function we used the rule {(Input Dimension * Hidden Layer) + (Hidden Layer * Output Layer) + Hidden Layer + Output Layer.

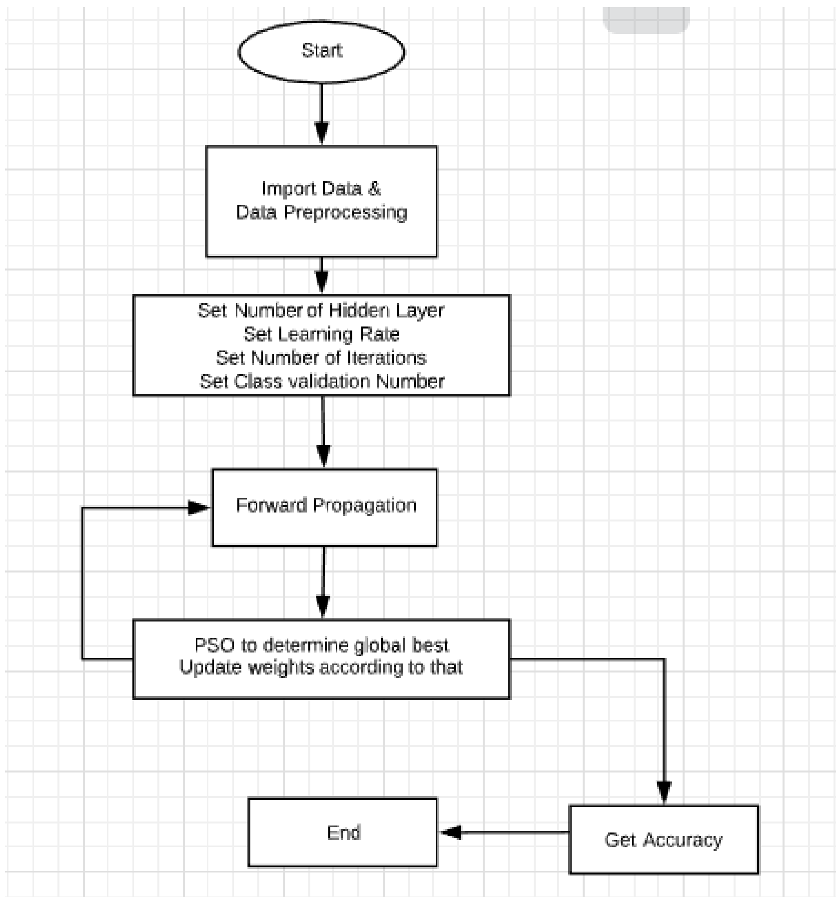

Figure 4.4: Flowchart of ANN-PSO

For other machine learning techniques such as Naïve Bayes, SVM, Random Forest, Nearest Centroid, Logistic Regression, K Neighbor the one hot encoding was used and the training data was applied to 70% and testing data was set to 30%.

Chapter 5

Experimental Results and Comparisons

5.1.1 ANN with Gradient Descent:

For Neural Network optimized by the gradient descent optimizer which is the most commonly used optimizing technique for neural network, we used N fold cross validation. As we used 4 folds there was four different confusion matrix.

The spilt was (Train = 1653, Test= 550).

For fold 1:

	Fatal	Serious	Slight
Fatal	0	4	5
Serious	0	38	197
Slight	0	11	1398

Table 5.1: Confusion Matrix

(Train Set)

	Fatal	Serious	Slight
Fatal	0	1	5
Serious	0	3	71
Slight	0	12	458

Table 5.2: Confusion Matrix

(Test Set)

	Precision	Recall	f1-score	Support
0	0.00	0.00	0.00	9
1	0.72	0.16	0.26	235
2	0.87	0.99	0.93	1409
avg / total	0.85	0.87	0.83	1653

Table 5.3: Precision, Recall, f1-score, Support for Train Set

	Precision	Recall	f1-score	Support
0	0.00	0.00	0.00	5
1	0.50	0.10	0.16	83
2	0.85	0.10	0.91	462
avg / total	0.79	0.84	0.79	550

Table 5.4: Precision, Recall, f1-score, Support for Test Set

Accuracy Train = 86.87% **Accuracy Test = 83.82%**

For fold 2:

	Fatal	Serious	Slight
Fatal	0	2	10
Serious	0	46	181
Slight	0	17	1397

Table 5.5: Confusion Matrix

(Train Set)

	Fatal	Serious	Slight
Fatal	0	2	1
Serious	0	8	74
Slight	0	17	448

Table 5.6: Confusion Matrix

(Test Set)

	Precision	Recall	f1-score	Support
0	0.00	0.00	0.00	12
1	0.71	0.20	0.32	227
2	0.88	0.99	0.93	1414
avg / total	0.85	0.87	0.84	1653

Table 5.7: Precision, Recall, f1-score, Support for Train Set

	Precision	Recall	f1-score	Support
0	0.00	0.00	0.00	3
1	0.30	0.10	0.15	82
2	0.86	0.96	0.91	465
avg / total	0.77	0.83	0.79	550

Table 5.8: Precision, Recall, f1-score, Support for Test Set

Accuracy Train = 87.30% **Accuracy Test = 82.91%**

For fold 3:

	Fatal	Serious	Slight
Fatal	0	1	13
Serious	0	66	173
Slight	0	20	1380

Table 5.9: Confusion Matrix (Train Set)

	Fatal	Serious	Slight
Fatal	0	0	1
Serious	0	6	64
Slight	0	20	459

Table 5.10: Confusion Matrix (Test Set)

	Precision	Recall	f1-score	Support
0	0.00	0.00	0.00	14
1	0.76	0.28	0.40	239
2	0.88	0.99	0.93	1400
avg / total	0.86	0.87	0.85	1653

Table 5.11: Precision, Recall, f1-score, Support for Train Set

	Precision	Recall	f1-score	Support
0	0.00	0.00	0.00	1
1	0.23	0.09	0.13	70
2	0.88	0.96	0.92	479
avg / total	0.79	0.85	0.81	550

Table 5.12: Precision, Recall, f1-score, Support for Test Set

Accuracy Train = 87.48% **Accuracy Test = 84.55%**

For fold 4:

	Fatal	Serious	Slight
Fatal	0	1	9
Serious	0	43	183
Slight	0	13	1404

Table 5.13: Confusion Matrix (Train Set)

	Fatal	Serious	Slight
Fatal	0	0	1
Serious	0	8	75
Slight	0	8	454

Table 5.14: Confusion Matrix (Test Set)

	Precision	Recall	f1-score	Support
0	0.00	0.00	0.00	10
1	0.75	0.19	0.30	226
2	0.88	0.99	0.93	1417
avg / total	0.86	0.88	0.84	1653

Table 5.15: Precision, Recall, f1-score, Support for Train Set

	Precision	Recall	f1-score	Support
0	0.00	0.00	0.00	5
1	0.50	0.10	0.16	83
2	0.85	0.98	0.91	462
avg / total	0.79	0.84	0.79	550

Table 5.16: Precision, Recall, f1-score, Support for Test Set

Accuracy Train = 87.54% **Accuracy Test = 84.00%**

Average Training Accuracy = 87.30%

Average Testing Accuracy = 83.82%

Total accuracy = 85.56%

Average Precision = 0.82

Average Recall = 0.86

Average f1-score =0.82

Total execution time = 319.82seconds

5.1.2 ANN with PCA:

We turned the dimension of the attributes from 12 to 2 using PCA.

The spilt was (Train = 1653, Test= 550).

For fold 1:

	Fatal	Serious	Slight
Fatal	0	4	5
Serious	0	21	221
Slight	0	7	1395

Table 5.17: Confusion Matrix (Train Set)

	Fatal	Serious	Slight
Fatal	0	1	5
Serious	0	5	75
Slight	0	7	457

Table 5.18: Confusion Matrix (Test Set)

	Precision	Recall	f1-score	Support
0	0.00	0.00	0.00	9
1	0.65	0.08	0.14	242
2	0.86	0.99	0.92	1402
avg / total	0.76	0.71	0.69	1653

Table 5.19: Precision, Recall, f1-score, Support for Train Set

	Precision	Recall	f1-score	Support
0	0.00	0.00	0.00	6
1	0.38	0.06	0.10	80
2	0.85	0.98	0.91	464
avg / total	0.81	0.71	0.76	550

Table 5.20: Precision, Recall, f1-score, Support for Test Set

Accuracy Train = 86.66%

Accuracy Test = 84.00%

For fold 2:

	Fatal	Serious	Slight
Fatal	0	4	9
Serious	0	41	195
Slight	0	13	1391

Table 5.21: Confusion Matrix (Train Set)

	Fatal	Serious	Slight
Fatal	0	1	5
Serious	0	5	74
Slight	0	7	458

Table 5.22: Confusion Matrix (Test Set)

	Precision	Recall	f1-score	Support
0	0.00	0.00	0.00	13
1	0.70	0.17	0.27	236
2	0.87	0.99	0.92	1404
avg / total	0.85	0.87	0.84	1653

Table 5.23: Precision, Recall, f1-score, Support for Train Set

	Precision	Recall	f1-score	Support
0	0.00	0.00	0.00	6
1	0.38	0.06	0.10	79
2	0.85	0.98	0.91	465
avg / total	0.77	0.83	0.79	550

Table 5.24: Precision, Recall, f1-score, Support for Test Set

Accuracy Train = 87.30% **Accuracy Test = 82.91%**

For fold 3:

	Fatal	Serious	Slight
Fatal	0	0	14
Serious	0	0	239
Slight	0	0	1400

Table 5.25: Confusion Matrix (Train Set)

	Fatal	Serious	Slight
Fatal	0	0	1
Serious	0	0	70
Slight	0	0	479

Table 5.26: Confusion Matrix (Test Set)

	Precision	Recall	f1-score	Support
0	0.00	0.00	0.00	10
1	0.00	0.00	0.00	226
2	0.86	0.86	0.79	1417
avg / total	0.73	0.86	0.79	1653

Table 5.27: Precision, Recall, f1-score, Support for Train Set

	Precision	Recall	f1-score	Support
0	0.00	0.00	0.00	5
1	0.00	0.00	0.00	83
2	0.84	1.00	0.91	462
avg / total	0.71	0.84	0.77	550

Table 5.28: Precision, Recall, f1-score, Support for Test Set

Accuracy Train = 85.30%

Accuracy Test = 85.27%

For fold 4:

	Fatal	Serious	Slight
Fatal	0	11	27
Serious	0	30	220
Slight	0	15	1350

Table 5.29: Confusion Matrix (Train Set)

	Fatal	Serious	Slight
Fatal	0	0	5
Serious	0	8	75
Slight	0	8	454

Table 5.30: Confusion Matrix (Test Set)

	Precision	Recall	f1-score	Support
0	0.00	0.00	0.00	38
1	0.53	0.12	0.19	250
2	0.84	0.99	0.90	1366
avg / total	0.75	0.79	0.81	1653

Table 5.31: Precision, Recall, f1-score, Support for Train Set

	Precision	Recall	f1-score	Support
0	0.00	0.00	0.00	5
1	0.50	0.10	0.16	83
2	0.85	0.98	0.91	462
avg / total	0.79	0.84	0.79	550

Table 5.32: Precision, Recall, f1-score, Support for Test Set

Accuracy Train = 83.54%

Accuracy Test = 84.00%

Average Training Accuracy = 85.70%

Average Testing Accuracy = 84.04%

Total accuracy = 84.87%

Average Precision =0.77

Average Recall = 0.81

Average f1-score=0.78

Total execution time = 197.30 seconds

5.1.3 ANN with GA:

In a regular basis back propagation is used in ANN to optimize the weights but instead of back propagation, we used genetic algorithm to update the weights.

	Fatal	Serious	Slight
Fatal	0	7	12
Serious	0	41	192
Slight	0	10	1391

Table 5.33: Confusion Matrix

	Precision	Recall	f1-score	Support
0	0.00	0.00	0.00	19
1	0.70	0.18	0.28	233
2	0.87	0.99	0.92	1401
avg / total	0.82	0.81	0.83	2203

Table 5.34: Precision, Recall, f1-score, Support

Accuracy = 86.63%

Execution time = 685.07 seconds

5.1.4 ANN with PSO:

ANN is optimized by the Particle Swarm Optimizer. The optimizing technique was used to update the weights in back propagation.

	Fatal	Serious	Slight
Fatal	0	0	23
Serious	0	35	304
Slight	0	7	1834

Table 5.35: Confusion Matrix

	Precision	Recall	f1-score	Support
0	0.00	0.00	0.00	23
1	0.83	0.14	0.24	339
2	0.85	0.99	0.91	1841
avg / total	0.82	0.81	0.83	2203

Table 5.36: Precision, Recall, f1-score, Support

Accuracy: 85.06%

5.1.5 Support Vector Machine (SVM):

	Fatal	Serious	Slight
Fatal	1	0	5
Serious	0	4	89
Slight	1	1	560

Table 5.37: Confusion Matrix

	Precision	Recall	f1-score	Support
0	0.50	0.17	0.25	6
1	0.80	0.04	0.08	93
2	0.86	1.00	0.92	562
avg / total	0.85	0.85	0.80	661

Table 5.38: Precision, Recall, f1-score, Support

Accuracy: 85.47%

5.1.6 Naïve Bayes:

	Fatal	Serious	Slight
Fatal	0	0	6
Serious	0	9	84
Slight	0	19	543

Table 5.39: Confusion Matrix

	Precision	Recall	f1-score	Support
0	0.00	0.00	0.00	6
1	0.32	0.10	0.15	93
2	0.86	0.97	0.91	562
avg / total	0.77	0.84	0.79	661

Table 5.40: Precision, Recall, f1-score, Support

Accuracy: 84.00%

5.1.7 Nearest Centroid:

	Fatal	Serious	Slight
Fatal	2	3	1
Serious	9	56	28
Slight	70	186	306

Table 5.41: Confusion Matrix

	Precision	Recall	f1-score	Support
0	0.02	0.33	0.05	6
1	0.23	0.60	0.33	93
2	0.91	0.54	0.68	562
avg / total	0.81	0.55	0.63	661

Table 5.42: Precision, Recall, f1-score, Support

Accuracy: 55.00%

5.1.8 Logistic Regression:

	Fatal	Serious	Slight
Fatal	0	0	6
Serious	0	1	92
Slight	0	0	562

Table 5.43: Confusion Matrix

	Precision	Recall	f1-score	Support
0	0.00	0.00	0.00	6
1	1.00	0.01	0.02	93
2	0.85	1.00	0.92	562
avg / total	0.86	0.85	0.79	661

Table 5.44: Precision, Recall, f1-score, Support

Accuracy: 85.17%

5.1.9 K Nearest Neighbors Classifier:

	Fatal	Serious	Slight
Fatal	1	0	5
Serious	1	18	74
Slight	4	80	478

Table 5.45: Confusion Matrix

	Precision	Recall	f1-score	Support
0	0.17	0.17	0.17	6
1	0.18	0.19	0.19	93
2	0.86	0.85	0.85	562
avg / total	0.76	0.75	0.75	661

Table 5.46: Precision, Recall, f1-score, Support

Accuracy: 75.18%

5.1.10 Random Forest:

	Fatal	Serious	Slight
Fatal	1	0	5
Serious	0	9	84
Slight	1	6	555

Table 5.47: Confusion Matrix

	Precision	Recall	f1-score	Support
0	0.50	0.17	0.25	6
1	0.60	0.10	0.17	93
2	0.86	0.99	0.92	562
avg / total	0.82	0.85	0.81	661

Table 5.48: Precision, Recall, f1-score, Support

Accuracy: 85.04 %

Name of Algorithms	Accuracy
GA-ANN	86.63%
ANN with Gradient Descent	85.56%
SVM	85.47%
Logistic Regression	85.17%
PSO-ANN	85.06%
Random Forest	85.04%
PCA-ANN	84.87%
Naïve Bayes	84.00%
K Nearest Neighbor	75.18%
Nearest Centroid	55.00%

Table 5.49: Accuracy Table of different techniques

Name of Algorithms	Precision	Recall	F1-Score
ANN with Gradient Descent	0.82	0.86	0.82
PCA-ANN	0.77	0.81	0.78
GA-ANN	0.82	0.81	0.83
PSO-ANN	0.82	0.81	0.83
SVM	0.85	0.85	0.80
Nearest Centroid	0.81	0.55	0.63
Logistic Regression	0.86	0.85	0.79
K Nearest Neighbor	0.76	0.75	0.75
Random Forest	0.82	0.85	0.81
Naïve Bayes	0.77	0.84	0.79

Table 5.50: Precision, Recall, F1-Score

5.2 Performance Evaluation:

We found that the best accuracy is obtained by the Genetic Algorithm – Artificial Neural Network hybridization technique which is 86.63%. The lowest accuracy achieved by Nearest Centroid method which is 55.00%. ANN-Gradient Descent performed well as well predicting with an accuracy of 85.56% which is the second best to GA-ANN technique. SVM, Logistic Regression, ANN-PSO, Random Forest all had their accuracy over 85%. ANN-PCA predicted with an accuracy of 84.87% which is very much similar to ANN-Gradient Descent and GA-ANN (85.56% and 86.63% respectively). But in ANN- Gradient Descent and GA-ANN the execution time was much more than PCA-ANN. By converting those 12 attributes to 2, we got the result (319.82-197.30) =122.52 seconds faster than ANN-Gradient Descent and (685.07-197.30) = 487.77 seconds faster than GA-ANN.

In the case of precision, recall and f1-score, the highest precision score is 0.86 which is obtained by Logistic Regression, highest recall 0.85 obtained by three algorithms SVM, Logistic Regression and Random Forest respectively, highest f1-score 0.83 obtained by GA-ANN and ANN-PSO.

Chapter 6

Conclusion and Future Work

In this paper, different machine learning models were proposed to observe the result and find out which model is performing better. We applied our model on a road accident dataset and predicted the severity. There were three different hybridization of Artificial Neural Network were implemented and in total ten different models proposed. Among them Genetic Algorithm – Artificial Neural Network hybridized model gave the best accuracy of 86.63%. On the other hand, PCA-ANN performed well and gave an accuracy of 84.87% but just took one third execution time of GA-ANN model. The models were ANN-Gradient Descent, ANN-PSO, Gaussian Naïve Bayes, Support Vector Machine, Random Forest, K Neighbor, Nearest Centroid and Logistic Regression.

There may be quite a few ways to improve our result in future. As neural network works better with large number of data, the first thing can be done is to add more data [18]. Missing values can be handled well as missing values cause the low prediction rate. Feature Engineering and feature selection can be applied to find out which attribute has greater impact on result and remove the attributes which has very less impact on result. Ensemble methods can also be applied in order to get better result [19]. These models can be applied on other areas and see if GA-ANN gives the best accuracy or not.

Chapter 7

Bibliography

[1] Q. A. Al-radaideh and E. J. Daoud, “Data Mining Methods for Traffic Accident Severity Prediction,” vol. 5, no. 2014, 2018.

[2] M. I. Sameen and B. Pradhan, “applied sciences Severity Prediction of Traffic Accidents with Recurrent Neural Networks,” 2017.

[3] M. M. Kunt, I. Aghayan, and N. Noii, “PREDICTION FOR TRAFFIC ACCIDENT SEVERITY : COMPARING THE ARTIFICIAL NEURAL NETWORK , GENETIC ALGORITHM , COMBINED GENETIC ALGORITHM AND PATTERN SEARCH METHODS,” vol. 26, no. 4, pp. 353–366, 2011.

[4] I. Learning, "Introduction to Gradient Descent Algorithm along its variants", *Analytics Vidhya*, 2019.[Online]. Available: https://www.analyticsvidhya.com/blog/2017/03/introduction-to-gradient-descent-algorithm-along-its-variants/. [Accessed: 29- Nov- 2019].

[5] S. Alkheder, M. Taamneh and S. Taamneh, "Severity Prediction of Traffic Accident Using an Artificial Neural Network", *Journal of Forecasting*, vol. 36, no. 1, pp. 100-108, 2016. Available: 10.1002/for.2425.

[6] *Naun.org*,2019.[Online].Available:http://www.naun.org/main/NAUN/neural/2018/a022016-069.pdf. [Accessed: 29- Nov- 2019].

[7] "Classification of brain tumors using PCA-ANN - IEEE Conference Publication", *Ieeexplore.ieee.org*,2019.[Online].Available:https://ieeexplore.ieee.org/document/6141398. [Accessed: 09- Dec- 2019].

[8] "Combinational Method for Face Recognition: Wavelet, PCA and ANN - IEEE Conference Publication", *Ieeexplore.ieee.org*,2019.[Online].Available:https://ieeexplore.ieee.org/document/4700005. [Accessed: 09- Dec- 2019].

[9] L. D. Zhang, L. Jia, and W. X. Zhu, “Overview of traffic flow hybrid ANN forecasting algorithm study,” *ICCASM 2010 - 2010 Int. Conf. Comput. Appl. Syst. Model. Proc.*, vol. 1, no. Iccasm, 2010.

[10] T. Hendtlass, “Particle swarm optimisation and high dimensional problem spaces,” 2009 IEEE Congr. Evol. Comput. CEC 2009, pp. 1988–1994, 2009.

[11] R. Jiménez, E. Gervilla, A. Sesé, J. J. Montaño, B. Cajal, and A. Palmer, “Dimensionality reduction in data mining using artificial neural networks,” Methodology, vol. 5, no. 1, pp. 26–34, 2009.

[12] L. T. Le, H. Nguyen, J. Dou, and J. Zhou, “A comparative study of PSO-ANN, GA-ANN, ICA-ANN, and ABC-ANN in estimating the heating load of buildings’ energy efficiency for smart city planning,” Appl. Sci., vol. 9, no. 13, 2019.

[13] "Genetic Algorithm-Artificial Neural Network (GA-ANN) Hybrid Intelligence for Cancer Diagnosis - IEEE Conference Publication", *Ieeexplore.ieee.org*, 2019. [Online]. Available: https://ieeexplore.ieee.org/document/5614106. [Accessed: 09- Dec- 2019].

[14] "Optimisation and Classification of EMG signal using PSO-ANN - IEEE Conference Publication", *Ieeexplore.ieee.org*,2019.[Online].Available:https://ieeexplore.ieee.org/document/8783882/references#references. [Accessed: 09- Dec- 2019].

[15] C. O. Sakar, S. O. Polat, M. Katircioglu, and Y. Kastro, “Real-time prediction of online shoppers’ purchasing intention using multilayer perceptron and LSTM recurrent neural networks,” Neural Comput. Appl., vol. 31, no. 10, pp. 6893–6908, 2019.

[16] J. Srisuan and A. Hanskunatai, “The ensemble of Naïve Bayes classifiers for hotel searching,” 2014 Int. Comput. Sci. Eng. Conf. ICSEC 2014, no. 1, pp. 168–173, 2014.

[17] Q. Sun, W. He, and L. Chen, “Multi-Label Automatic Labeling for Question Attributes Based on Adaboost and Bayes Algorithms,” Proc. 2018 Chinese Autom. Congr. CAC 2018, pp. 2955–2960, 2019.

[18] "How to improve performance of Neural Networks", *d4datascience.wordpress.com*, 2019. [Online]. Available: https://d4datascience.wordpress.com/2016/09/29/fbf/. [Accessed: 08- Dec- 2019].

[19] 8. Model, "8 Proven Ways for boosting the "Accuracy" of a Machine Learning Model", *AnalyticsVidhya*,2019.[Online].Available:https://www.analyticsvidhya.com/blog/2015/12/improve-machine-learning-results/. [Accessed: 08- Dec- 2019].

[20] K. L. S. Dl, W. V. Frp, Z. Qdqndl, and H. G. X. Fq, “$ *$ $11 0Rgho Iru $Lu 4Xdolw\ 3Uhglfwlqj,” pp. 1–7, 2010.

[21] F. Ahmad, N. A. Mat-Isa, Z. Hussain, R. Boudville, and M. K. Osman, “Genetic Algorithm - Artificial Neural Network (GA-ANN) hybrid intelligence for cancer diagnosis,” *Proc. - 2nd Int. Conf. Comput. Intell. Commun. Syst. Networks, CICSyN 2010*, pp. 78–83, 2010, doi: 10.1109/CICSyN.2010.46.

[22] N. Pugh and H. Park, “Prediction of Red-Light Running using an Artificial Neural Network,” *Conf. Proc. - IEEE SOUTHEASTCON*, vol. 2018-April, pp. 1–4, 2018, doi: 10.1109/SECON.2018.8479222.