International Journal of E-Health and Medical Communications
Volume 12 • Issue 4 • Bi-Monthly 2021

Performance Analysis of Machine Learning Algorithms for Big Data Classification:

ML and AI-Based Algorithms for Big Data Analysis

Sanjeev Kumar Punia, JIMS Engineering Management Technical Campus, India

Manoj Kumar, School of Computer Science, University of Petroleum and Energy Studies (UPES), Dehradun, India

https://orcid.org/0000-0001-5113-0639

Thompson Stephan, Department of Computer Science and Engineering, Faculty of Engineering and Technology, M. S. Ramaiah University of Applied Sciences, Bangalore,Noida, India

Ganesh Gopal Deverajan, Galgotias University, India

https://orcid.org/0000-0003-0036-7841

Rizwan Patan, Velagapudi Ramakrishna Siddhartha Engineering College, India

https://orcid.org/0000-0003-4878-1988

ABSTRACT

In broad, three machine learning classification algorithms are used to discover correlations, hidden patterns, and other useful information from different data sets known as big data. Today, Twitter, Facebook, Instagram, and many other social media networks are used to collect the unstructured data. The conversion of unstructured data into structured data or meaningful information is a very tedious task. The different machine learning classification algorithms are used to convert unstructured data into structured data. In this paper, the authors first collect the unstructured research data from a frequently used social media network (i.e., Twitter) by using a Twitter application program interface (API) stream. Secondly, they implement different machine classification algorithms (supervised, unsupervised, and reinforcement) like decision trees (DT), neural networks (NN), support vector machines (SVM), naive Bayes (NB), linear regression (LR), and k-nearest neighbor (K-NN) from the collected research data set. The comparison of different machine learning classification algorithms is concluded.

KEYWORDS

Big Data, Decision Trees (DT), Floating Centroid Method (FCM), Neural Networks (NN), Semantic Web Service Classification, Spark Framework, Support Vector Machines (SVM), Twitter Streaming API

1. INTRODUCTION

In the current digital era, data is growing exponentially. The amount of this growing data known as Big Data is the beginning of the human life revolution in many fields. In general, the five main characteristics of Big Data are (i) volume (ii) variety (iii) velocity (iv) veracity and (v) value. The combination of these five characteristics is called 5 Vs. and is represented in Figure 1, Where "volume" represents the collection of all generated data sets. The "variety" indicates the different formats of

DOI: 10.4018/IJEHMC.20210701.oa4

data from various sources. The “velocity” shows the high speed of accumulation of data in the data set. The “veracity” represents data accuracy or trustworthiness in the generated data set. The “value” represents all types of attributes in the generated data set. Big data analysis is growing rapidly in every field/industry. In medical science, big data analysis is used to prevent and cure different diseases like cancer. Big data analyses benefit hospitals by providing better patients satisfaction. In the field of agriculture, the analysis of big data helps to increase agriculture product value. In space-related research, big data analysis provides many opportunities in exploring different researches. In pattern recognition, big data analyses play a vital role during remote sensing. The use of big data has already given rise to several questions, including those of how data can be collected and used in ethical and socially sensitive ways.

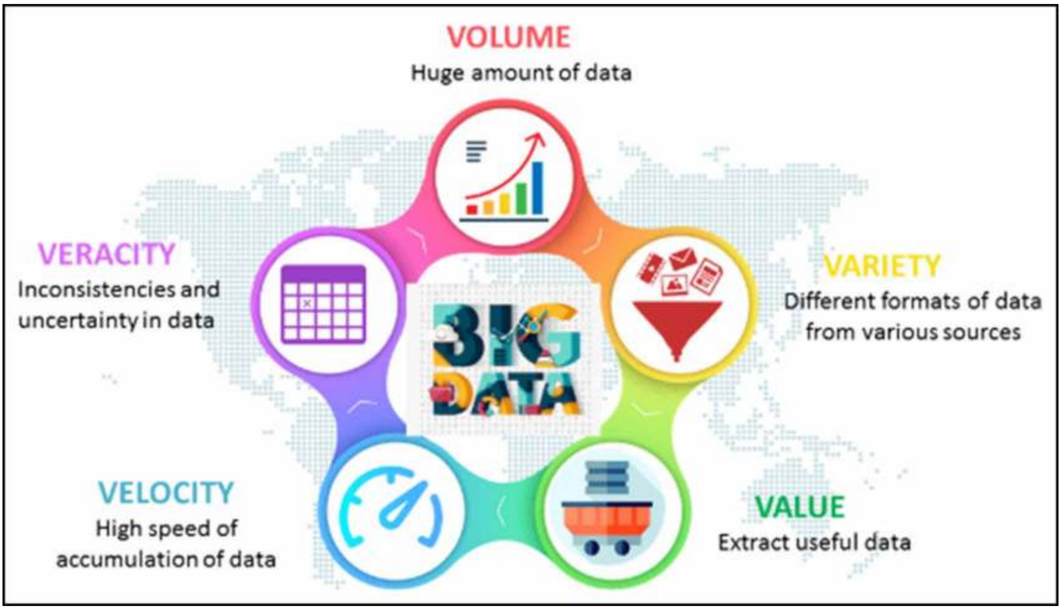
In this paper, section II describes different classification techniques. Section III represents a classification literature survey. Section IV displays the experimental setup. Section V shows the result analysis and section VI concludes the paper with its limitation.

1.1 Classification of Techniques

In this paper, we used five different classifications algorithms for big data analysis, namely (i) Decision Trees (DT) (ii) Neural Networks (NN) (iii) Support Vector Machines (SVM) (iv) Naive Bayes (NB), and (v) k-Nearest Neighbor (K-NN) classification algorithms. Dana and Alashqur (2014) explained that the Decision Tree (DT) classification algorithm is based on a tree-like structure. The main characteristics of decision tree classification algorithms are (a) designing a problem such that it is easy to understand (b) reducing the complexity of the problem. The main disadvantage of decision trees is that they are unstable, which means a minor change in the data can lead to a dramatic structural change in the optimal decision tree.

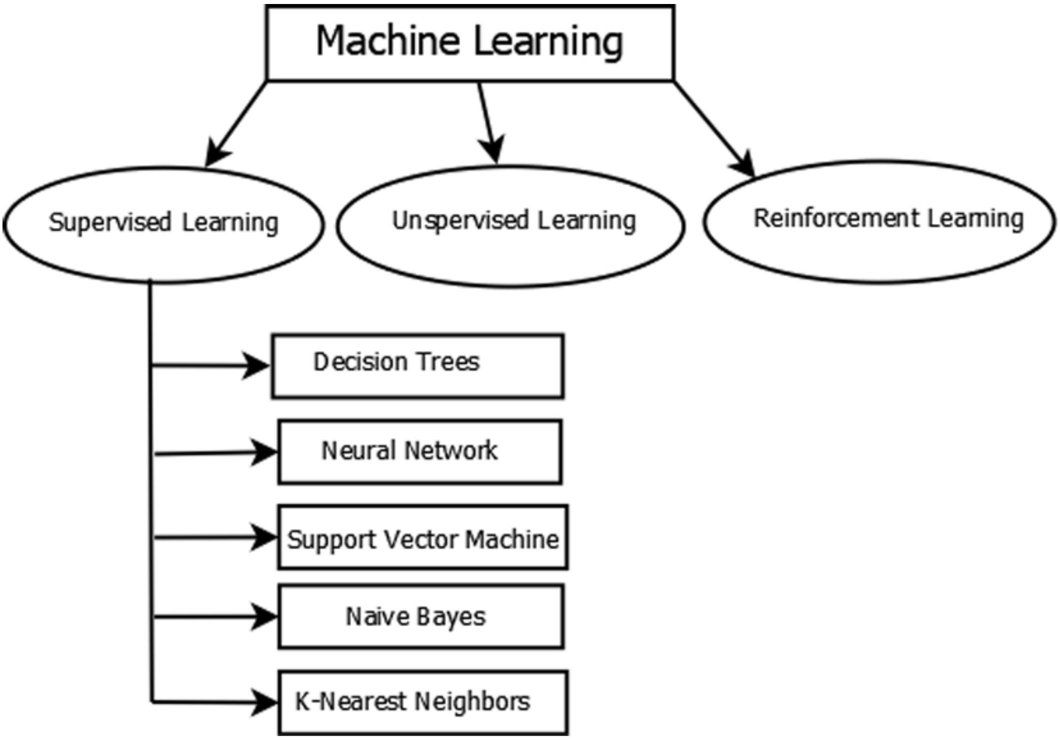
Unnikrishnan et al. (2017) explained that the Neural Network (NN) classification algorithm works as a biological artificial neuron perception and receives different input according to the features. The neural network classification algorithm sums all these input features and produces the final result accordingly. The major advantage of the neural network classification algorithm is to handle the

Figure 1. Big Data five v’s


VOLUME
Huge amount of data
VARIETY
Different formats of data from various sources
VALUE
Extract useful data
VELOCITY
High speed of accumulation of data
VERACITY
Inconsistencies and uncertainty in data
BIG DATA

unhandled linear programming task. The major disadvantage of the neural network classification algorithm is the high processing time requirement, especially in large neural networks. Romero et al. (2010) explained that the Support Vector Machine (SVM) classification algorithm is derived from statistical learning theory. Support Vector Machines is a supervised learning algorithm currently used in machine learning to classify large data sets. Support Vector Machines can accurately map pre-defined social categories with relevant industry, which helps identify key themes being discussed on social media. The SVM classification algorithm's disadvantage is that it takes more time to train the collected data for high accuracy prediction.

Figure 2. Machines learning classification algorithms



Bhardwaj et al. (2019) explained that the Naive Bayes (NB) classification algorithms are based on Bayes' Theorem. It is a probabilistic machine learning model which is used for classifying task. The major advantage of the Naive Bayes classification algorithm is easy building for very large datasets with no complicated iterative parameter estimation. The major disadvantage of the Naive Bayes classification algorithm is the requirement of independent predictors. The Naive Bayes classification algorithm performance is based on different real-life cases. Mohammed et al. (2018) explained that the k-Nearest Neighbors (KNN) classification algorithm implements efficiently with a supervised machine learning algorithm used to solve both classification and regression problems. The k-Nearest Neighbors classification algorithms are widely used in industry-based classification problems. It stores all available cases and performs the classification of new cases according to a similarity measure. In real-life scenarios, it is widely disposable as it does not assume data distribution; i.e., it is non-parametric.

2. RELATED WORKS

Grover and Johari (2015) suggested big data collection process categories with their complexities in different environments. The authors describe the process of storage, retrieval, archive, and handling real-time commercial transaction data and the limitations of the relational database, which forms the roots of big data. Further, the authors showed the complete working procedure of different big data tools like NoSQL and MongoDB. Hashem and Ranc (2016) designed a relational database management system to overcome the limitations of big data generation collection. Reddy and Kumar (2016) proposed a semantic exploitation healthcare system to integrate collected big data from different sources. The authors described various big data models and process engines for health care integration. Giraldo et al. (2008) proposed a model to study respiratory variability patterns for weaning trials patients based on SVM classification. The main reason to design a mechanical ventilator is to breathe the patient easily. They proposed a model to study the differences in patients' respiratory patterns based on weaning and proved that the SVM method is the best for the patient's respiratory pattern. Hassan and Bermak (2014) proposed a gas classification with sensors to challenge real-life applications. The binary decision tree approach for gas classification based on sensors' sensitivities difference is used. The pairs of sensors split the available gas data samples at the decision node in two branches. At decision node, a single sensor pair is selected based on a distance metric. The authors conclude that gas performance degrades by implementing different pattern recognition algorithms with different concentrations.

Kaur and Bhagla (2016) proposed the Naïve Bayes classification model for the Hindi language to disambiguate to extract different nouns, collocation, local context, and unordered word list. The authors demonstrate that the Naïve Bayer classification model efficiency increases by adding more features in an unordered word list. Liu et al. (2019) designed a clinical decision support system based on different advanced data mining techniques to improve clinical decisions. These systems increase diagnostic accuracy by decreasing diagnosis time. The proposed model stores the patient historical data in clouds using cryptographic techniques by preserving individual patient clinical data. Alty et al. (2018) proposed a patient arterial stiffness model for cardiovascular disease without patient blood. The patient's volume pulse is measured by placing an infrared light absorption detector on the patient index finger. The authors proved that more than 92% accuracy could be predicted using SVM on waveform extracted features. Jiang et al. (2019) proposed the Naive Bayes classifiers method for the Chinese language to categorize text simplicity, tokens, etc. The Naïve Bayes classification model predicts results using conditional probability based on different weighting approach. Liu et al. (2018) proposed a Naive Bayes web-based service classification technique for semantic web services. The Naive Bayes semantic web service approach processed the user interaction quickly and accurately. The authors elaborate on the concrete process of Naive Bayes semantic web services classifier to enhance service efficiency.

Liu et al. (2019) proposed an innovative neural network floating centroid method (FCM) approach to predict the share market trend with high accuracy. The authors extracted real-time and off-line stock market data to analyze and visualize and further explained the influence of different stock market characteristics on share prices based on traditional neural network algorithms. Abou Elassad et al. (2020) proved that predicting the stock market is based on the random selection's initial weight that can be easily prone to incorrect predictions.

3. EXPERIMENTAL SETUP

Apache Spark (an open-source) framework is the modern big data processing engine that offers faster solutions than the Hadoop MapReduce technique, especially beneficial for the machine learning classification algorithm. Apache Spark framework supports clustering, classification, reduction, regression, etc., to perform various tasks like data analytics, machine learning, data streaming, database management, parallel computing, graph operations. Apache Spark framework supports Java, Scala, Python, and R. We use Spark ML Lib (Machine learning libraries) sometimes written by MLibthat provides a wide range of advanced machine learning libraries. Spark can process a large amount of data and perform advanced machine learning algorithms on it.

In this model, first, we build a pipeline to process the real-time Twitter data using the Apache Spark framework. In the next step, we fetched real-time Twitter data using Twitter (tweepy) streaming API and stored it in JSON Objects. This JSON object contains the tweets, user-details, re-tweets, IP address, etc. We processed the tweet and re-tweet only using Apache-Spark in our experiment.

The different steps to process Twitter live streaming data are (i) first, set up a pipeline to send the request to the server (ii) download real-time Twitter data (iii) store Twitter live streaming data (iv) accept the developer agreement to create the access tokens (v) set up a pipeline to send Twitter live stream data to Apache Spark framework (vi) process stored Twitter live stream data with Spark framework and (vii) finally, close the server connection.

In our model, we process and analyzed the live twitter stream stored data through five machine learning classification algorithms as (i) Decision Trees (DT) (ii)Neural Networks (NN) (iii) Support Vector Machines (SVM) (iv) Naive Bayes (NB) and (v) k-Nearest Neighbor (K-NN) classification algorithms. The different stages of our model are shown in Figure 3.

Twitter's live stream data is retrieved in a file through a Twitter streaming application programming interface. The complete steps for twitter's retrieval process are given below and shown in Figure 4.

During Twitter live stream data collection and process, we perform the following steps -

Step 1: Stream Twitter data using a Spark package called **Twitter.utils** that contains all the built-in functions to stream data from Twitter.

Figure 3. System Process Stages

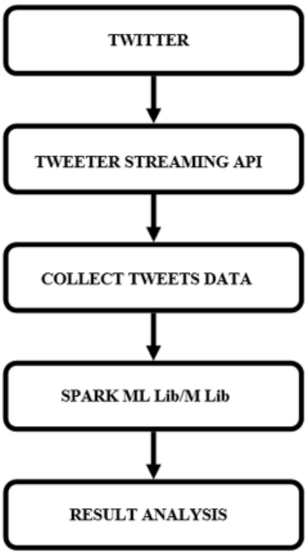
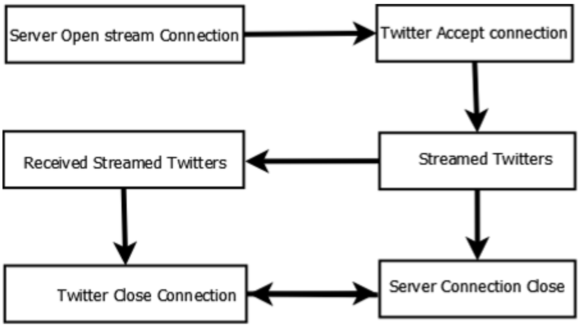
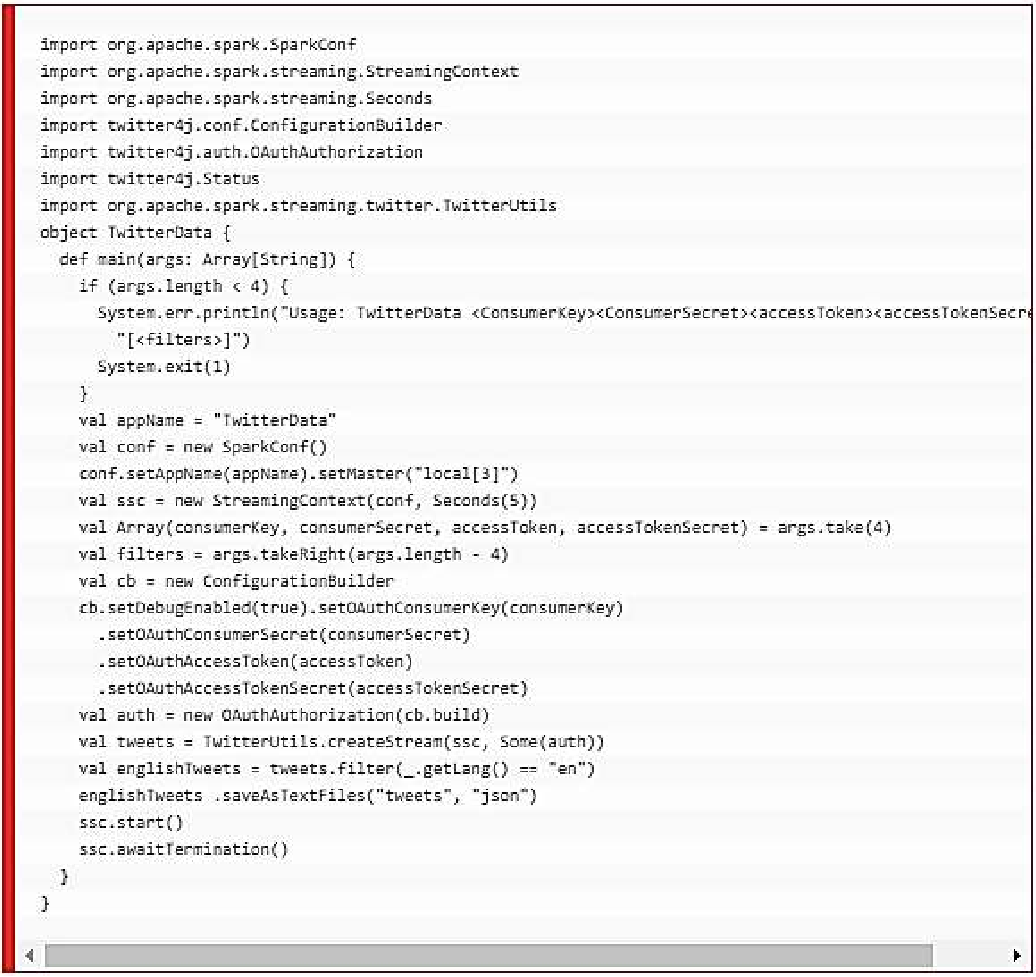


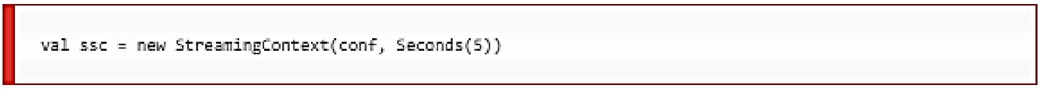
Figure 4. Twitters retrieval process


Server Open stream Connection
Twitter Accept connection
Received Streamed Twitters
Streamed Twitters
Twitter Close Connection
Server Connection Close



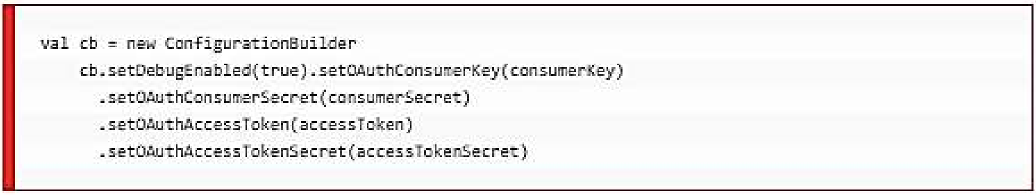
```
import org.apache.spark.SparkConf
import org.apache.spark.streaming.StreamingContext
import org.apache.spark.streaming.Seconds
import twitter4j.conf.ConfigurationBuilder
import twitter4j.auth.OAuthAuthorization
import twitter4j.Status
import org.apache.spark.streaming.twitter.TwitterUtils
object TwitterData {
  def main(args: Array[String]) {
    if (args.length < 4) {
      System.err.println("Usage: TwitterData <ConsumerKey><ConsumerSecret><accessToken><accessTokenSecre
        "[<filters>]")
      System.exit(1)
    }
    val appName = "TwitterData"
    val conf = new SparkConf()
    conf.setAppName(appName).setMaster("local[3]")
    val ssc = new StreamingContext(conf, Seconds(5))
    val Array(consumerKey, consumerSecret, accessToken, accessTokenSecret) = args.take(4)
    val filters = args.takeRight(args.length - 4)
    val cb = new ConfigurationBuilder
    cb.setDebugEnabled(true).setOAuthConsumerKey(consumerKey)
      .setOAuthConsumerSecret(consumerSecret)
      .setOAuthAccessToken(accessToken)
      .setOAuthAccessTokenSecret(accessTokenSecret)
    val auth = new OAuthAuthorization(cb.build)
    val tweets = TwitterUtils.createStream(ssc, Some(auth))
    val englishTweets = tweets.filter(_.getLang() == "en")
    englishTweets .saveAsTextFiles("tweets", "json")
    ssc.start()
    ssc.awaitTermination()
  }
}
```

Step 2: Set the Spark streaming context as follows:



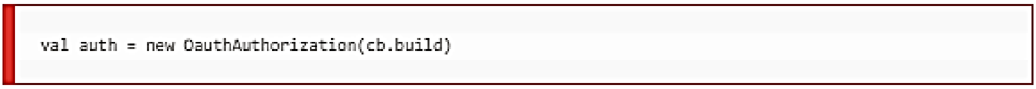
```
val ssc = new StreamingContext(conf, Seconds(5))
```

Step 3: Use the Configuration Builder class to take the keys for Twitter authentication as follows:



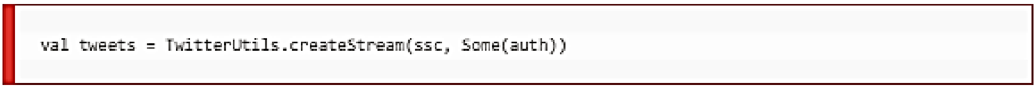
```
val cb = new ConfigurationBuilder
    cb.setDebugEnabled(true).setOAuthConsumerKey(consumerKey)
      .setOAuthConsumerSecret(consumerSecret)
      .setOAuthAccessToken(accessToken)
      .setOAuthAccessTokenSecret(accessTokenSecret)
```

Step 4: Perform authorization as follows:



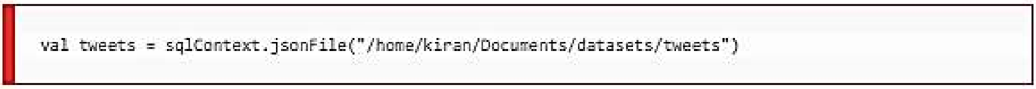
```
val auth = new OauthAuthorization(cb.build)
```

Step 5: Start spark streaming using the TwitterUtils.createStream class as follows.



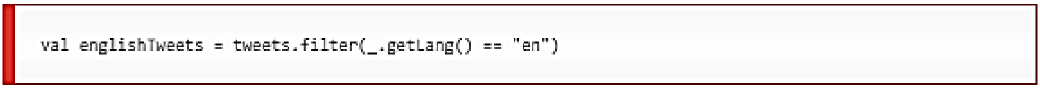
```
val tweets = TwitterUtils.createStream(ssc, Some(auth))
```

Step 6: Streaming application stream the data and store it in the variable **tweets** in JSON format.
Step 8: Filter the **English** languages tweets only as follows



```
val tweets = sqlContext.jsonFile("/home/kiran/Documents/datasets/tweets")
```

Step 9: The Filtered tweets based on language is added to the **Twitter-4j-3.0.6** jar and processed.



```
val englishTweets = tweets.filter(_.getLang() == "en")
```

4. RESULT AND PERFORMANCE ANALYSIS

The result is calculated based on different sensitivity parameters present in the collected research data. Initially, we separate the collected research data into two different research groups based on true and false data values. Later, we divide each research group into two sub research groups based on positive and negative data values. A True positive is an outcome where the model correctly predicts the positive class. Similarly, a true negative is an outcome where the model correctly predicts the negative class. In the first condition (true data value), the representation is shown below:

- Total True Positive (TP): Total real positive values present in true research group data.
- Total True Negative (TN): Total real negative values present in true research group data.

A false positive is an outcome where the model incorrectly predicts the positive class. Moreover, a false negative is an outcome where the model incorrectly predicts the negative class. In the second condition (false data value), the representation is shown below:

- Total False Positive (FP): Total real positive values present in false research group data.
- Total False Negative (FN): Total real negative values present in false research group data.

The True Positive Rate (TPR) predicts the actual positive instances in true research group data and calculated using equation 1.

$$\mathrm{TPR} = \frac{\mathrm{TP}}{\left(\mathrm{TP} + \mathrm{TN}\right)} \tag{1}$$

The True Negative Rate (TNR) predicts the actual negative instances in true research group data and calculated using equation 2.

$$\mathrm{TNR} = \frac{\mathrm{TN}}{\left(\mathrm{TP} + \mathrm{TN}\right)} \tag{2}$$

The False Positive Rate (FPR) predicts the actual positive instances in false research group data and calculated using equation 3.

$$\mathrm{FPR} = \frac{\mathrm{FP}}{\left(\mathrm{FP} + \mathrm{FN}\right)} \tag{3}$$

The False Negative Rate (FNR) predicts the actual negative instances in false research group data and calculated using equation 4.

$$\mathrm{FNR} = \frac{\mathrm{FN}}{\left(\mathrm{FP} + \mathrm{FN}\right)} \tag{4}$$

Accuracy assumes equal costs for both kinds of errors. The 99% accuracy can be excellent or terrible depending upon the problem. The overall accuracy is calculated by summing the number of correctly predicted values and dividing by the total number of predicted values as represented in equation 5.

$$\mathrm{Accuracy} == \frac{\left(\mathrm{TP} + \mathrm{TN}\right)}{\left(\mathrm{TP} + \mathrm{FP} + \mathrm{TN} + \mathrm{FN}\right)} \tag{5}$$

We plot the curve for five different classification algorithms (Decision Trees, Neural Networks, Support Vector Machines, Naive Bayes and k-Nearest Neighbor) based on different sensitivity (true positive rate, true negative rate, false positive rate, false negative rate, and accuracy) and data set size relation. Finally, we compare all five classification algorithms based on their sensitivity values with respective data set size. The curves perform Graph processing using Graph Xcomponent as shown.

4.1 True Positive Rate (TPR)

The comparison of True Positive Rate (TPR) curve in Figure 5 shows that TPR values 0.3, 0.5, 0.7, 0.9 and 0.95 corresponding to data set size 5000, 10000, 30000, 50000 and 60000 respectively are highest in Support Vector Machine (SVM) classification algorithm among all five classification algorithms. Next, TPR values 0.25, 0.53, 0.7, 0.75 and 0.85 corresponding to data set size 5000, 20000, 40000, 50000 and 60000 respectively are higher in Naive Bayes (NB) classification algorithm among the rest four classification algorithms. Hence, the above result shows that the SVM classification algorithm is best followed by the Naive Bayes (NB) classification algorithm.

Figure 5. True Positive Rate (TPR)

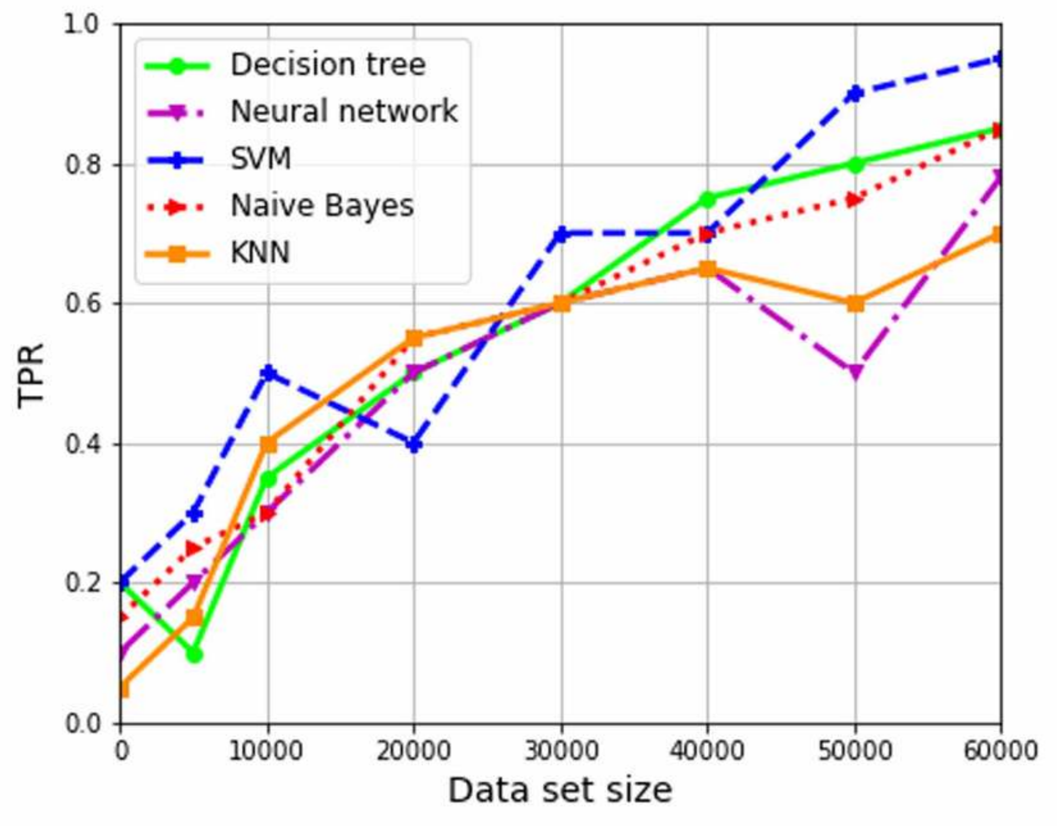


Table 1. True Positive Rate (TPR) Values

Model & data	Decision Tree	Neural Network	SVM	Naive Bayes	KNN
0	0.2	0.1	0.2	0.15	0.05
5000	0.1	0.2	0.3	0.25	0.15
10000	0.35	0.3	0.5	0.3	0.4
20000	0.5	0.5	0.4	0.55	0.55
30000	0.6	0.6	0.7	0.6	0.6
40000	0.75	0.65	0.7	0.7	0.65
50000	0.8	0.5	0.9	0.75	0.6
60000	0.85	0.78	0.95	0.85	0.7

4.2 True Negative Rate (TNR)

The comparison of True Negative Rate (TNR) curves in Figure 6 shows that TNR values 0.3, 0.6, 0.9, 0.75, 0.75 and 0.8 corresponding to data set size 5000, 20000, 30000, 40000, 50000 and 60000 respectively are highest in Naive Bayes (NB) classification algorithm among five classification algorithms. Next, TNR values of 0.3, 0.5, 0.8, and 0.7 correspond to the data set size 5000, 10000,

Table 2. True Negative Rate (TNR) Values

Model & data	Decision Tree	Neural Network	SVM	Naive Bayes	KNN
0	0.3	0.1	0.2	0.15	0.08
5000	0.4	0.2	0.3	0.3	0.15
10000	0.1	0.4	0.5	0.2	0.45
20000	0.6	0.5	0.3	0.6	0.2
30000	0.4	0.6	0.8	0.9	0.5
40000	0.7	0.6	0.6	0.75	0.6
50000	0.75	0.5	0.7	0.75	0.62
60000	0.9	0.75	0.6	0.8	0.7

30000, and 50000, respectively, are higher in the SVM classification algorithm among the rest of the four classification algorithms. Hence, the above result shows that the SVM classification algorithm best follows naive Bayes (NB) classification algorithm.

4.3 False Positive Rate (TNR)

The comparison of False Positive Rate (FPR) curves in Figure 7 shows that FPR values 0.3, 0.6, 0.9, 0.75, 0.75 and 0.9 corresponding to the data set size 5000, 20000, 30000, 40000, 50000 and 60000, respectively are highest in Naive Bayes (NB) classification algorithm among the five classification algorithms. Next, FPR values of 0.3, 0.5, 0.8, and 0.7 correspond to data set size 5000, 10000, 30000and 50000 are higher in the SVM classification algorithm among the rest of the four classification algorithms. Hence, the above result shows that the SVM classification algorithm best follows naive Bayes (NB) classification algorithm.

4.4 False Negative Rate (FNR)

The comparison of False Negative Rate (FNR) curve in Figure 8 shows that FNR values are 0.35, 0.4, 0.7, 0.72 and 0.9, corresponding to data set size 5000, 10000, 30000, 50000 and 60000, respectively. The FNR value is highest in the Support Vector Machine (SVM) classification algorithm among all five classification algorithms. Next, FNR values 0.25, 0.53, 0.6, 0.8, 0.75 and 0.7 corresponding to data set size 5000, 20000, 30000, 40000, 50000 and 60000, respectively are higher in Naive Bayes (NB) classification algorithm among rest four classification algorithms. Hence, the above result shows that the SVM classification algorithm is best followed by the Naive Bayes (NB) classification algorithm.

4.5. Accuracy

The Accuracy comparison curve in Figure 9 shows that accuracy values are 0.3, 0.4, 0.65, 0.7, 0.7, 0.9 and 0.95 corresponding to data set size 5000, 10000, 20000, 30000, 40000, 50000 and 60000 respectively. The accuracy value is highest in the SVM classification algorithm among all five classification algorithms. Next, accuracy values 0.25, 0.3, 0.55, 0.6, 0.7, 0.8 and 0.85 corresponding to data set size 5000, 10000, 20000, 30000, 40000, 50000 and 60000, respectively are higher in Naive Bayes (NB) classification algorithm among rest four classification algorithms. The SVM classification algorithm's accuracy values are always higher throughout the accuracy comparison curve, followed by the Naive Bayes (NB) classification algorithm. From the above results, it can be concluded that the SVM classification algorithm is best followed by the Naive Bayes (NB) classification algorithm.

Figure 6. True Negative Rate (TNR) 3. False Positive Rate (TNR)

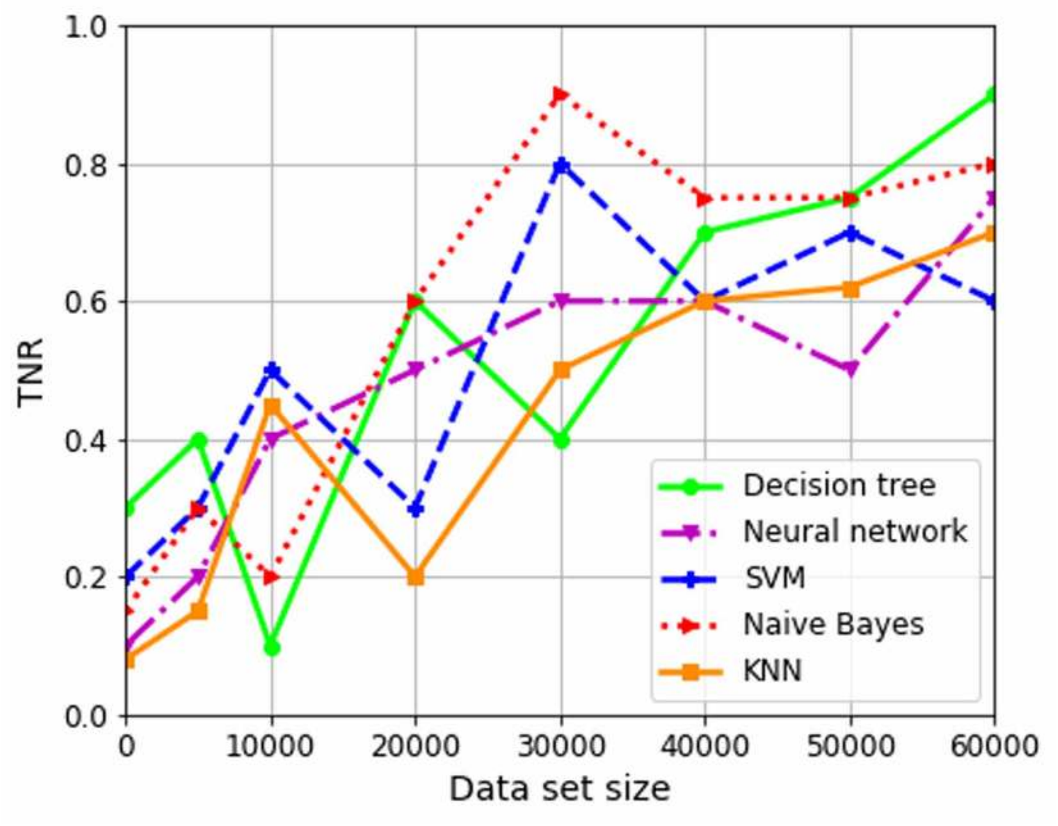


Table 3. False positive rate values

Model &Data	Decision Tree	Neural Network	SVM	Naive Bayes	KNN
0	0.3	0.1	0.2	0.15	0.07
5000	0.4	0.2	0.35	0.3	0.14
10000	0.1	0.4	0.5	0.2	0.45
20000	0.6	0.5	0.3	0.6	0.2
30000	0.4	0.6	0.8	0.9	0.5
40000	0.7	0.7	0.6	0.75	0.56
50000	0.75	0.5	0.7	0.75	0.6
60000	0.9	0.75	0.6	0.9	0.7

5. CONCLUSION AND FUTURE ASPECTS

The result shows that the average values of accuracy for Decision Trees, Neural Networks, SVM, Naive Bayes, and k-Nearest Neighbor are 0.3875, 0.4625, 0.6, 0.525 and 0.37, respectively. Hence, this paper concludes that SVM classification algorithm yields the best classification accuracies among

Figure 7. False Positive Rate (FPR)

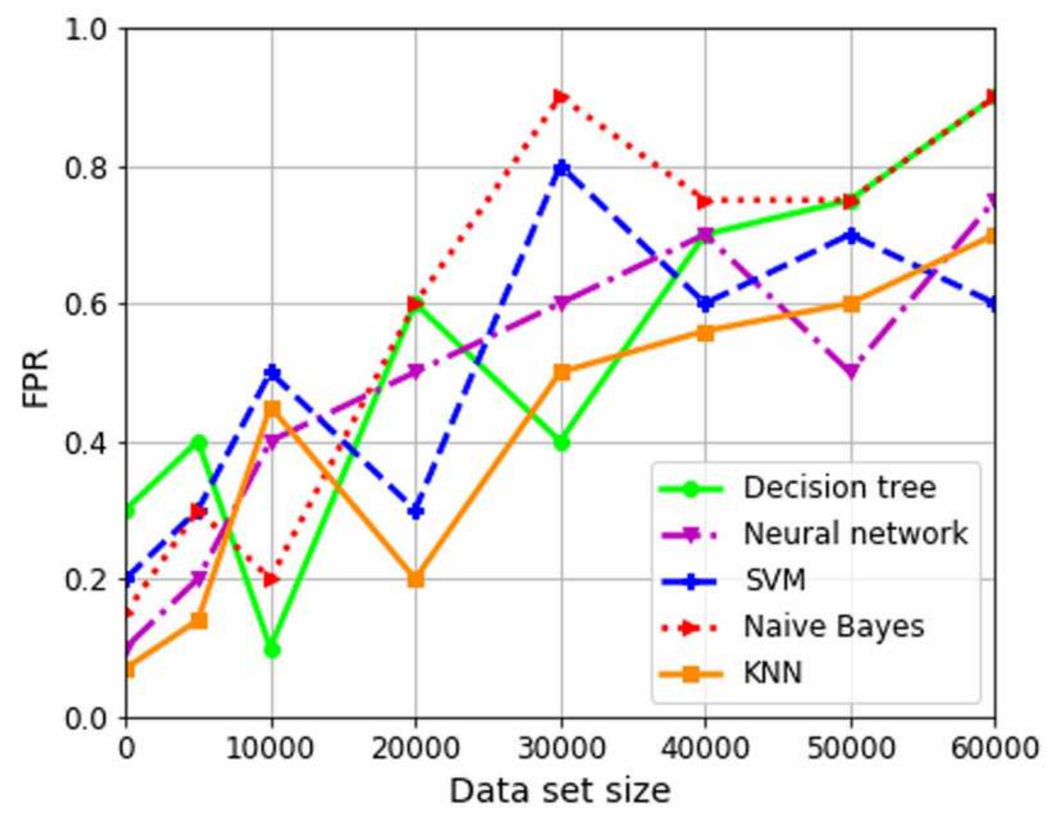


Table 4. False Negative Rate (FNR) Values

Model & data	Decision Tree	Neural Network	SVM	Naive Bayes	KNN
0	0.2	0.1	0.2	0.15	0.05
5000	0.1	0.2	0.35	0.25	0.15
10000	0.35	0.4	0.4	0.3	0.32
20000	0.4	0.5	0.3	0.55	0.4
30000	0.3	0.6	0.7	0.6	0.3
40000	0.4	0.7	0.6	0.8	0.4
50000	0.6	0.5	0.7	0.75	0.6
60000	0.4	0.75	0.9	0.7	0.4

five discussed classification algorithms as Decision Trees, Neural Networks, SVM, Naive Bayes, and k-Nearest Neighbor followed by Naive Bayes (NB) classification algorithm based on different sensitivity values (True Positive, True Negative, False Positive, and False Negative). The results could be enhanced by using different kernels for different classification algorithms for the same data

Figure 8. False Negative Rate (FNR)

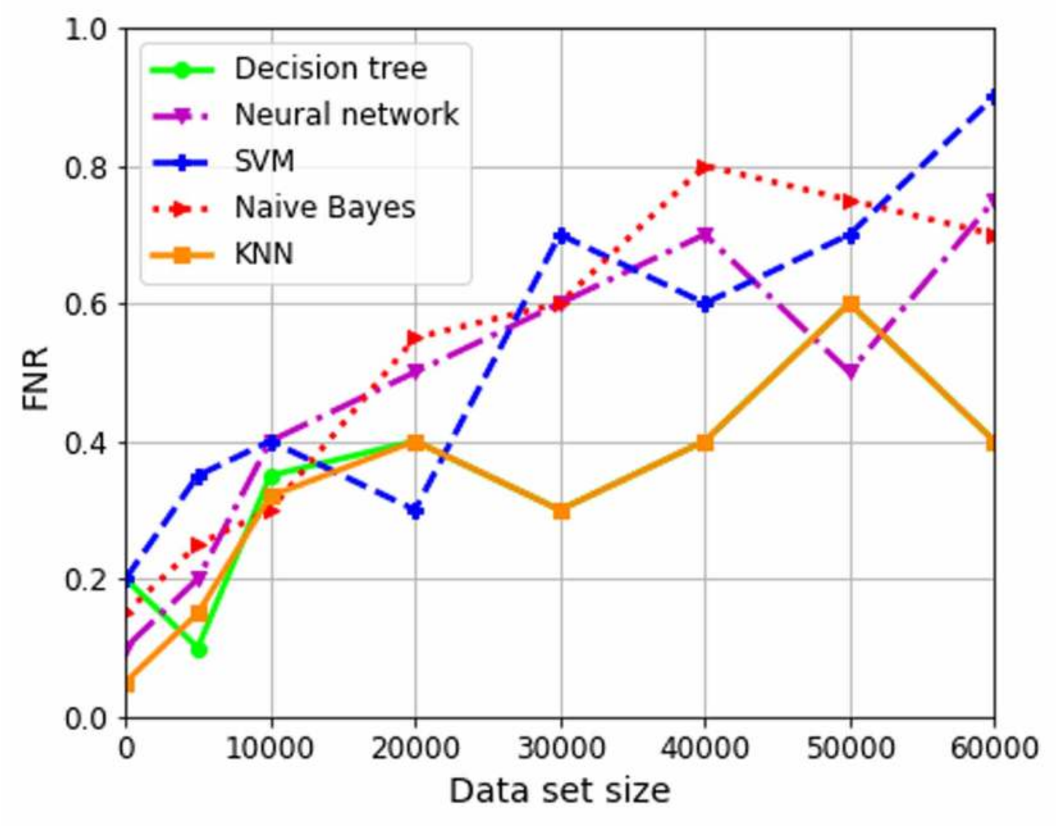
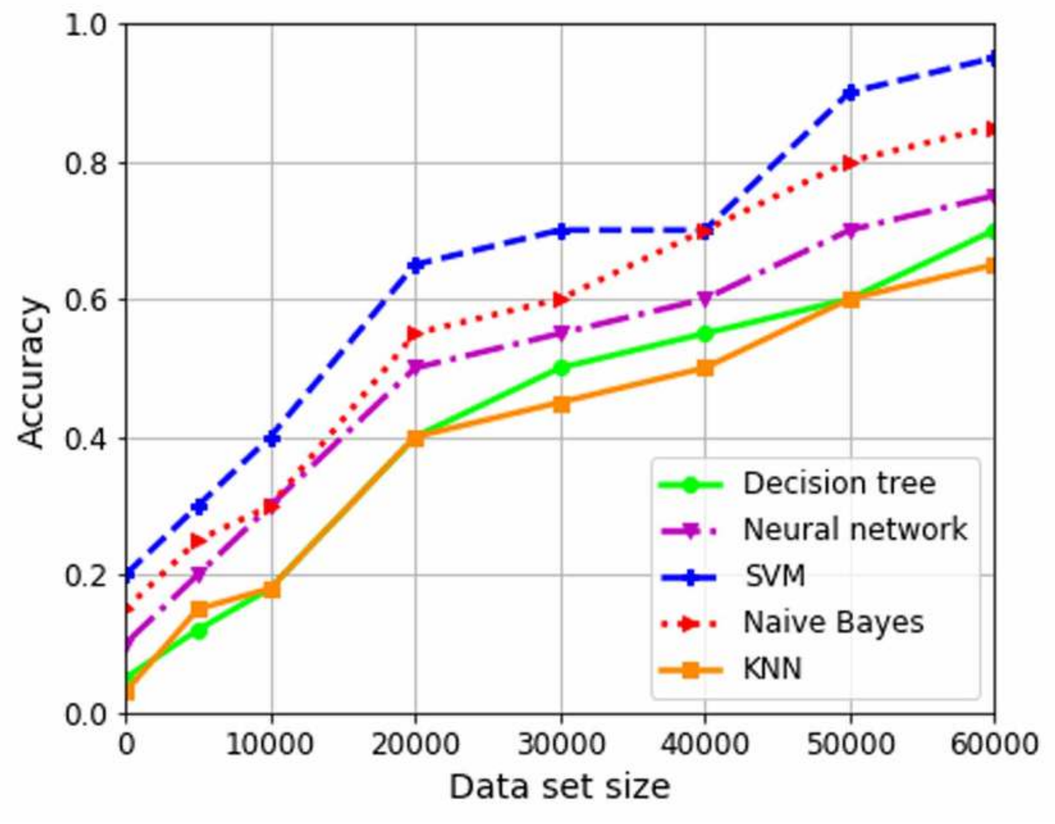


Table 5. Accuracy values

Decision Tree	Neural Network	SVM	Naive Bayes	KNN
0.05	0.1	0.2	0.15	0.03
0.12	0.2	0.3	0.25	0.15
0.18	0.3	0.4	0.3	0.18
0.4	0.5	0.65	0.55	0.4
0.5	0.55	0.7	0.6	0.45
0.55	0.6	0.7	0.7	0.5
0.6	0.7	0.9	0.8	0.6
0.7	0.75	0.95	0.85	0.65

set. Similarly, a random forest could generate different results while splitting a node by changing the number of decision trees within the underlying forest.

Figure 9. Accuracy



REFERENCES

Abou Elassad, Z. E., Mousannif, H., Al Moatassime, H., & Karkouch, A. (2020). The application of machine learning techniques for driving behavior analysis: A conceptual framework and a systematic literature review. *Engineering Applications of Artificial Intelligence*, *87*, 103312.

Alty, S. R., Millasseau, S. C., Chowienczyc, P. J., & Jakobsson, A. (2003, December). Cardiovascular disease prediction using support vector machines. In *2003 46th Midwest Symposium on Circuits and Systems* (Vol. 1, pp. 376-379). IEEE.

Bhardwaj, A., Singh, V. K., & Narayan, Y. (2015, December). Analyzing BigData with Hadoop Cluster in HDInsight Azure Cloud. In *2015 Annual IEEE India Conference (INDICON)* (pp. 1-5). IEEE.

Dana, A. D., & Alashqur, A. (2014, March). Using decision tree classification to assist in the prediction of Alzheimer's disease. In *2014 6th International Conference on Computer Science and Information Technology (CSIT)* (pp. 122-126). IEEE.

Giraldo, B. F., Garde, A., Arizmendi, C., Jane, R., Diaz, I., & Benito, S. (2008). Support vector machine classification applied on weaning trials patients. In *Encyclopedia of Healthcare Information Systems* (pp. 1277–1282). IGI Global.

Grover, P., & Johari, R. (2015, April). BCD: BigData, cloud computing and distributed computing. In *2015 Global Conference on Communication Technologies (GCCT)* (pp. 772-776). IEEE.

Hashem, H., & Ranc, D. (2016, May). A review of modeling toolbox for BigData. In *2016 International Conference on Military Communications and Information Systems (ICMCIS)* (pp. 1-6). IEEE.

Hassan, M., & Bermak, A. (2014, June). Gas classification using binary decision tree classifier. In *2014 IEEE International Symposium on Circuits and Systems (ISCAS)* (pp. 2579-2582). IEEE.

Jiang, Q., Wang, W., Han, X., Zhang, S., Wang, X., & Wang, C. (2016, August). Deep feature weighting in Naive Bayes for Chinese text classification. In *2016 4th International Conference on Cloud Computing and Intelligence Systems (CCIS)* (pp. 160-164). IEEE.

Kaur, J., & Bhagla, S. (2016). News Classification Using Naïve Baye's Classifier. *International Journal of Advanced Research in Computer Science and Software Engineering*, *6*(4), 698–702.

Liu, J., Tian, Z., Liu, P., Jiang, J., & Li, Z. (2016, June). An approach of semantic web service classification based on Naive Bayes. In *2016 IEEE International Conference on Services Computing (SCC)* (pp. 356-362). IEEE.

Liu, S., Yang, B., Wang, L., Zhao, X., Zhou, J., & Guo, J. (2016, August). Prediction of share price trend using FCM neural network classifier. In *2016 3rd International Conference on Informative and Cybernetics for Computational Social Systems (ICCSS)* (pp. 81-86). IEEE.

Liu, X., Lu, R., Ma, J., Chen, L., & Qin, B. (2015). Privacy-preserving patient-centric clinical decision support system on naive Bayesian classification. *IEEE Journal of Biomedical and Health Informatics*, *20*(2), 655–668. doi:10.1109/JBHI.2015.2407157 PMID:26960216

Mohammed, A. F., Humbe, V. T., & Chowhan, S. S. (2016, February). A review of big data environment and its related technologies. In *2016 International Conference on Information Communication and Embedded Systems (ICICES)* (pp. 1-5). IEEE.

Reddy, A. R., & Kumar, P. S. (2016, February). Predictive big data analytics in healthcare. In *2016 Second International Conference on Computational Intelligence & Communication Technology (CICT)* (pp. 623-626). IEEE.

Romero, C., Valdez, M. G., & Alanis, A. (2010, July). A comparative study of machine learning techniques in blog comments spam filtering. In *The 2010 International Joint Conference on Neural Networks (IJCNN)* (pp. 1-7). IEEE.

Unnikrishnan, A., Narayanan, U., & Joseph, S. (2017, August). Performance analysis of various supervised algorithms on big data. In *2017 International Conference on Energy, Communication, Data Analytics and Soft Computing (ICECDS)* (pp. 2293-2298). IEEE. doi:10.1109/ICECDS.2017.8389861

Ganesh Gopal received his PhD degree in Computer Science and Engineering from VIT University, Vellore, India, 2015. He has more than 13 years of Research and Teaching Experience in the domain of Computer Science and Engineering. Currently he is a Professor in School of Computing Science & Engineering, Galgotias University, India. He is currently supervising 5 M.Tech students and guiding 3 PhD students. His research interest includes Internet of Things (IoT), Wireless Communication, Vehicular Communication and Big Data. He has received from VIT University a research award as one of the top performer for 3 consecutive years 2015, 2016 & 2017. He has published papers in several international conferences and journals. He has contributed his ideas and shared his research experience as Technical Programme Committee (TPC) member, Programme Committee member and Session Chair member in premium international conferences. He has given key note addresses in many international conferences He has acted as accreditation consultant for ABET, BRICS and QS world ranking for several institutions across India. He has edited many special issues in reputed journals. He is a reviewer in some of the Q1 Journals. He has involved in several professional activities and as a member of professional committees like IEEE, ACM and CSI.