Biomedical Research 2017; Special Issue: S236-S241 *ISSN 0970-938X*
www.biomedres.info

An improved firefly heuristics for efficient feature selection and its application in big data.

Senthamil Selvi R[1*], Valarmathi ML[2]

[1]MIET College, Tiruchirappalli, Tamil Nadu, India

[2]Government College of Technology, Coimbatore, Tamil Nadu, India

Abstract

Big Data is exceedingly useful for business applications and is fast rising as a domain of the IT industry. It has created considerable interest in several domains, which includes the manufacturing of health care machine, bank transaction, social media, and so on. Due to the diversity and size of datasets in Big Data, effective representation, access as well as analyses of unstructured as well as semi-structured data are still problematic. It is required to determine the way of searching space of all potential variable sub-sets as well as the assessment of prediction performance of learning machines for guiding searches and also which predictor to utilize. Extensive searches may be carried out if the quantity of parameters is not too much. However the issue is NP-Hard and search rapidly turns operationally intractable. Vast set of search schemes may be utilized, which include best-first, branch-and-bound, simulated annealing, genetic algorithm. In the current paper, a features selection method on the basis of Firefly Algorithm (FA) is suggested to improve the big data analysis. FA meta-heuristic techniques modelled on the behaviour of the fireflies solve the optimization problems. The suggested technique was tested through a huge twitter data set and effectiveness of the proposed method was proven.

Keywords: Big data, Feature Selection, NP-hard, Firefly Algorithm (FA).

Accepted on February 02, 2017

Introduction

The notion of Big Data has been prevalent in the domain of computer science from the beginning. It initially was utilized for referring to huge quantities of data which was not capable of being processed in an efficient manner through conventional dataset techniques and tools. Every time, a novel storage medium was introduced, the quantity of accessible data increased as well, as it was easily accessible. The initial definition had a focus on structured data, however almost all research scholars as well as experts understand that most of the current data available in present in an unstructured format, primarily in the format of text or images. The explosive growth in the quantity of data was not accompanied by related rise of novel storage media. Big Data is understood as the vast quantity of data with a large variety of information which is beyond technology's capacity for storage, processing or management in an efficient manner [1].

The present growth rate of the quantity of data gathered is incredible. A huge problem faced by the IT research community as well as industry is that the rate exceeds the capacity to design adequate models for effective handling of data as well as analysis of the data for extraction of useful meaning assisting in making decisions. Big Data has three primary features, which are: 1) data is innumerous, 2) data is not capable of being sorted into regular, relational datasets, and 3) data is being created, obtained as well as processed in an extremely rapid manner. Big data as well as analyses on big data are at the core of modern science and business. With rapid growth in networking, data storage as well as data accumulation capabilities has increased exponentially, with big data is foraying into all fields of science and engineering, which includes physical, biological and bio-medical fields. Conventionally, data storage occurs in extremely structured format for maximization of information content. But, presently, data volume is mostly comprised of unstructured or semi-structured data. Hence, end-to-end processing is hindered by translations between structured data in relational systems of database management as well as unstructured data for analysis.

Applications like Google or Facebook on the web, deal with huge quantities of users which has never been taken into consideration by local applications. Size of datasets consumed by current web applications is too huge, particularly in the domains of finance, communication as well as business informatics due to the rise in the application of IT as well as online transactions across the Internet. In spite of the several problems associated with Big Data, there is no consensus on the quantification of Big Data, which relies on several factors, some of which are: 1) data structural complexity, 2) needs of target application. Problems in Big Data Analyses include lack of consistency, completeness, scalability, timeliness as well as

security [2]. Before data analysis, data is to be excellently structured. But because of the diversity of datasets in Big Data, effective representation, access as well as analyses of unstructured as well as semi-structured data are still problematic. Comprehending the techniques through which data may be pre-processed is significant for improving data quality as well as outcomes of analyses. Data sets are typically huge at several GigaBytes or more and they emerge in heterogeneous sources. Therefore, current real world datasets are extremely vulnerable to non-consistent, non-complete as well as noise-filled data. Hence, various data pre-processing methods, which include data cleaning, integrations, transformations as well as reductions are to be employed for removing noise as well as inconsistency [3,4]. Feature selection is critical in big data analysis to handle the numerous features in the data. Feature selection effectively removes redundant and irrelevant features obtaining an optimal feature subset which improves the efficacy of the classification and prediction algorithms. It also reduces the execution time required.

The focus of this investigation is to improve the efficacy of big data analysis using feature selection to obtain an optimal subset of features for analyzing the data. Features selection is utilized for two interconnected reasons: 1) bigger features sets result in greater variance in predictive modeling as well as comparatively increased overfitting and therein increased predictor errors, 2) in most applications, there are only a few relevant attributes amongst a huge set of non-relevant ones, wherein the non-relevant attributes increase variance as well as opportunity for overfitting, with no balance of learning improved models. Naïve Bayes Classifier (NBC) as well as K Nearest Neighbor (KNN) are utilized for achieving fine-grain control of analysis process for Hadoop implementations. The rest of the paper is structured thus. Section 2 presents detailed big data review. Section 3 summarizes proposed classifiers and optimization algorithms. Section 4 discussed in detail about the obtained results. Finally, Section 5 draws the conclusion.

Literature Survey

Big Data refers to collecting huge quantities of data sets such that it is too costly or complicated for processing through usage of generic data handling applications or social dataset administration models or desktop measurement as they require extremely huge parallel programming running on hundreds or thousands of servers. Problems include investigating, catching, inquiring, exchanging and so on. Sri and Anusha [5] studied Hadloop infrastructure, various tools utilized for Big Data analysis as well as related security issues. Bello-Orgaz et al. [6] suggested revisions to novel methods which were designed for permitting effective data mining as well as data fusion from social media as well as to novel applications as well as model which are currently emerging under the broad scope of social network, social media as well as Big Data paradigm. Big Data is concerned with huge volumes of complicated datasets with several autonomous sources. With rise in the growth of networking, data storage as well as data gathering capabilities, Big Data is now extending into all science and engineering fields, which includes physical, biological as well as bio-medical sciences. Wu et al. [7] suggested a HACE theorem which features the aspects of Big Data revolution and suggests a processing framework from the perspective of information mining. The data-driven model includes demand-driven collection of information sources, mining as well as analyses, user interest modeling, as well as security and privacy consideration. Problems in data-driven models and in the Big Data revolution were analysed.

Chen et al. [8] reviewed the background as well as current scene of Big Data. Generic background to Big Data as well as related technology (like cloud computing, Internet of Things and so on) were introduced. The focus then shifts to the 4 stages of value chain of big data, which include data creation, acquisition, storage as well as analyses. For every stage, generic background, technical problems as well as current progress were detailed. In the end, various representative applications of Big Data, which include enterprise management, Internet of Things, collective intelligence and so on were examined. The study was aimed at offering a comprehensive picture of the domain. The current trend, observed in almost every domain, is ebbing toward unparalleled growth in the quantity of data present in the world as well as the related opportunity and unexplored value. Chen et al. [9] reviewed Big Data problems from the perspective of data management. Particularly, Big Data diversity, integration, cleaning, indexing as well as analysis and mining were detailed. A short review on Big Data oriented research as well as issues were presented. Fortuny et al. [10] offers a clear picture of how Big Data is more useful for predictive analyses. This means that organizations with bigger datasets and the skill for exploiting them acquire considerable edge over the other organizations. Furthermore, the outcomes imply that it is possible for companies with access to fine-grained data, to collect more data samples as well as data attributes. An implementation of multi-variate Bernoulli Naïve Bayes protocols which is capable of scaling to huge, sparse data was also introduced.

Gao et al. [11] detailed the research effort into Big Data benchmarking with various partners in industries. However, access logs and related data come under business confidentiality that obstructs the construction of benchmarks. For overcoming such problems, open source solutions in search engines are studied and permission for utilizing anonymous Web Access Logs was got. Furthermore, with 2 years of effort, a semantic search engine called ProfSearch was built. This led to the creation of Big Data benchmark suite from search engines—BigDataBench, released on (http://prof.ict.ac.cn/BigDataBench). Detailed analyses on search engine workload as well as current benchmarking methods were reported. Novel data creation methods as well as tools were suggested for generating scalable quantities of Big Data from small quantities of real data, preservation of semantics as well as locality of data. Liu et al. [12] focused on the evaluation of scalability of Naïve Bayes Classifier in huge data sets. Rather than utilizing standard libraries, NBC was

implemented for achieving fine-grain control of analysis process. Outcomes are promising in that precision of NBC is enhanced and reaches 82% when dataset size rises. It was further proven that NBC is capable of scaling up for analyzing sentiments of millions of movie reviews with increased throughput.

Methodology

The focus of the work is on novel feature selection techniques and Firefly Optimization meta heuristic for solving the optimal subset selection problem. The selected features are classified using Naïve Bayes (NB), K Nearest Neighbor (KNN) and Multilayer Preceptron Neural Network (MLPNN).

Proposed firefly algorithm (FA) based feature selection

Firefly Algorithm (FA) is excellent for local searches however it might get forced into local optima and thereby cannot perform global searches well [13]. FA variables do not change with time during cycles. Two variables of the protocol are attractiveness as well as randomization coefficients. Values are crucial for determination of speed as well as behaviour of FA.

FA meta-heuristic function's on the principle of fireflies flashing lights for attracting one another. Light intensity assists firefly swarms in moving to brighter locations that are mapped to optimum solutions in search space.

The protocol standardizes certain Firefly characteristics which are:

- Fireflies are attracted to others, with no regard to sex.
- Fireflies' brightness is directly proportional to attractiveness. Fireflies with greater brightness attract those with lesser brightness. Fireflies move arbitrarily if they are not able to discover brighter fireflies.

Mathematically, fireflies' brightness have their basis in objective functions.

Attractiveness: In FA, attractiveness function of fireflies is monotonously decreasing functions specified by equation (1):

$$\beta(r) = \beta_0^* \exp(-\gamma r^m), \text{ with m} \geq 1 \rightarrow (1)$$

Wherein r represents distance between 2 fireflies, β_0 represents initial attractiveness at r=0, while γ represents an absorption coefficient that controls decreasing brightness.

Distance: Distance between 2 fireflies i as well as j, at positions x_i and x_j, correspondingly, are given as Cartesian/ Euclidean distance like in equation (2):

$$r_{ij} = \|x_i - x_j\| = \sqrt{\sum_{k=1}^{d} (x_{i,k} - x_{j,k})^2} \rightarrow (2)$$

Wherein $x_{i,k}$ represents k^{th} component of spatial coordinate x_i of i^{th} firefly while d represents quantity of dimensions, for d=2, equation (3):

$$r_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \rightarrow (3)$$

But, distance computation may be given through other distance measures on the basis of nature of the problem like Manhattan or Mahalanobis distance.

Movement: Movement of firefly i attracted by j is expressed as in equation (4):

$$x_i = x_i + \beta_0 * \exp(-\gamma r_{ij}^2) * (x_j - x_i) + a * (rand - \frac{1}{2}) \rightarrow (4)$$

Wherein the 1st term is current position of a firefly, 2nd term is firefly's attractiveness to light intensity as observed by adjacent fireflies while the 3rdterm is Firefly's arbitrary movement if there are no brighter ones. The coefficient α represents a randomization variable defined by a problem of interest, and rand represents an arbitrary number generated with uniform distribution in space [0,1]. FA meta-heuristic is selected for its capacity for providing optimum solutions for multi-objective issues.

The work flow is described as follows:

Dataset with T instances

T instances spilt into m maps

Feature Selection using Firefly algorithm

Input:

- Maximum number of Generation MaxGeneration,
- light absorption coefficient γ
- Objective function $f(x)$, $x=(x_1, \ldots, x_4)^T$

Initialize:

Generate initial population of fireflies $x_i(i=1,2,\ldots,n)$ where each firefly represents a feature

Light intensity I_i at x_i is determined by $f(x_i)$

While (t>MaxGeneration)

Move firefly based on I

- Calculate Attractiveness, Distantce, Movement
- Attractiveness varieties with distance r
- Evaluate new solution and update light intensity

Rank the fireflies and find the current best

End while

Optimal feature subset obtained

Classify the feature using Naïve Bayes, KNN and MLPNN

Implementation

A famous Big Data implementation environment is Google MapReduce. The model is constructed on the basis of two abstract functions which are Map as well as Reduce that are obtained from traditional functional programming paradigms. Consumers specify computations with regard to a map

(specifying per-record computations) as well as reduce (specifying result aggregations) functions that fulfill certain requisites. For instance, for supporting these, MapReduce needs the computations performed at reduce task to be associative as well as commutative. Various implementations of MapReduce have emerged in recent times with the most famous one being Apache Hadoop, an open-source model in Java which permits processing as well as management of huge data sets in distributed computational environment. Additionally, Hadoop functions on Hadoop Distributed File System (HDFS) that duplicates data files in several storage nodes, allowing fast information transfer rate amongst nodes and permitting system to continue operations with no interruptions when one or many nodes experience failure. Apache Hadoop is utilized for implementation. The block diagram of the suggested protocol utilized in MapReduce is given in Figures 1 and 2.

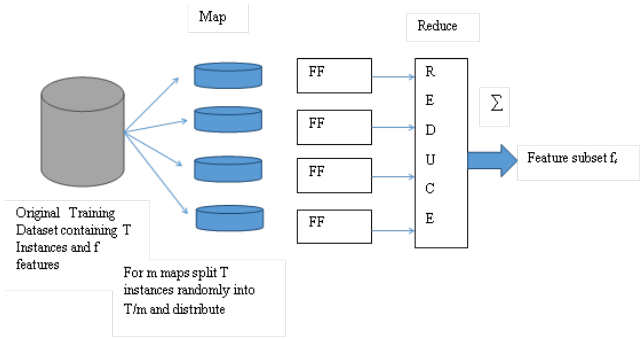

Figure 1. *Block diagram of the proposed algorithm.*

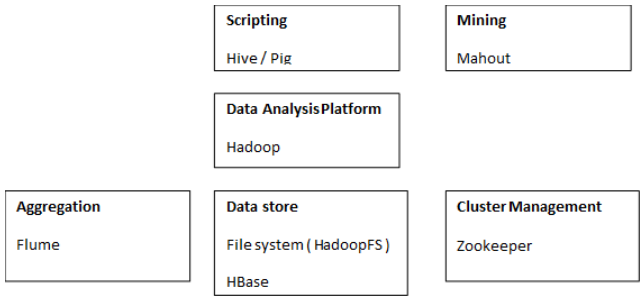

Figure 2. *Implementation architecture.*

Classifiers

Naïve bayes classification

Naïve Bayes has demonstrated excellent performance on sparse data sets. Excellent characteristic of Naïve Bayes is that it is very rapid in running on huge, sparse data sets when formulated right. Primary cause for the speed is because of the fact that Naïve Bayes ignores inter-feature dependency through assumption that within-class co-variances of features are 0. Naïve Bayes has demonstrated itself as a simple as well as efficient machine learning technique in earlier text classification works and as optimum in few situations (Liu et al., 2013).

Let there be m possible classes $C=c_1,c_2,....c_m$ for a set of documents $D=d_1,d_2,....,d_n$. Assume $W=w_1,w_2,...,w_s$ is a set of unique words, each of which appears a minimum of one time in one of the documents in D. The probability of a document d being in class c may be calculated through Bayes' rule:

$$P(c \mid d) = \frac{P(d \mid c)P(c)}{P(d)} \rightarrow (5)$$

AsP(d) is a constant for known dataset size, denominator of (1) is generally not computed for Maximum A Posteriori (MAP), typically for parametric statistical issues. A Naïve Bayes model presumes that every term or word, w_k, in a document occurs in an independent manner in the document given class c. Hence, (5) is:

$$P(c \mid d) \approx P(c) \prod_{k=1}^{n_d} [P(w_k \mid c)]^{t_k} \rightarrow (6)$$

Wherein n_d represents quantity of unique words in document d while t_k represents frequency of every word W_k. For avoiding floating point underflow, utilize (7):

$$\log P(c \mid d) \propto \log P(c) + \sum_{k=1}^{n_d} [t_k \log P(w_k \mid c)] \rightarrow (7)$$

Class of d is determined as class, c that performs maximization of $LogP(c|d)$ in (8):

$$c^* = \text{argmax}_c \in C \left\{ \log P(c) + \sum_{k=1}^{n_d} [t_k \log P(w_k \mid c)] \right\} \rightarrow (8)$$

When employing NBC, can predict P(c) as well as $P(w_k|c)$ like in (9):

$$\hat{P}(c) = \frac{N_c}{N} \text{ as well as } \hat{P}(w_k \mid c) = \frac{N_{w_k}}{\sum_{w_i \in W} N_{w_i}} \rightarrow (9)$$

wherein N represents total quantity of documents, Nc represents quantity of documents in class c while N_{w_i} represents frequency of a word wi in class c. With these estimates, computation of the right hand side of (3) is basically a counting problem. This ensures MapReduce is an adequate model for implementing NBC in huge data sets.

K-nearest neighbor (KNN)

Nearest-Neighbour Classifiers have their basis in learning through resemblance, which is to say, through comparison of specified test instance with training instance which is similar to it. It is a method which is utilized for identifying unknown data points on the basis of nearest neighbour whose values are already recognized, simple to comprehend however implausible work has in domains particularly in classification. Nearest neighbour classifiers have their basis in learning through analogy.

For data instance x to be sorted, k-nearest neighbours are searched and this makes neighbourhood of x. If k is too small,

then nearest-neighbor classifier is vulnerable to overfitting due to noise in training data. Whereas, if k is too big, nearest-neighbour classifier might mis-classify test samples as its list of closest neighbours might include data points which are situated far from the neighbourhood. KNN basically functions on the postulation that data is liked in features space. Therefore, every point is comprised in it, for finding out distance amongst points, Euclidian or Hamming distance is utilized as per the data type of data classes utilized [14]. Herein, one number k is specified that is utilized for finding total quantity of neighbours which determines the classification. IF value of k=1, then it is merely known as nearest neighbour. KNN needs: an integer k,a training data sample as well as a measure for assessing nearness.

Multi-layer perceptron neural network (MLPNN)

Multi-Layer Perceptron (MLP) is a particular ANN infrastructure. Let there be access to a training data set of l pairs $(\bar{x}_i, y_i)$ wherein $\bar{x}_i$ represents a vector comprising pattern, whereas y_i represents a class of associated pattern. For a 2-class task y_i may be encoded as +1 as well as −1. MLP possesses layers of nonlinear but distinguishable parametric functions. MLP is utilized with a single hidden layer; hidden as well as output layers possess tanh(.) transfer functions. MLPs are trained with gradient descent utilizing BP protocols for optimizing derivable criteria such as Mean Squared Error. Here, MLPs are trained for classifying inputs to be either specified clients or imposters. Inputs of MLPs are vectors that correspond to image extracted attributes. MLP outputs are either 1 or -1. MLPs are trained utilizing client as well as imposter images and thereby, MLP utilizes a discriminative training method.

MLP classifier utilizes the protocol given below for calculating input a node j obtained [15]:

$$net_j = \sum_i w_{ij} I_i$$

Wherein net_j represents input that one node j obtains; w_{ij} denotes weights between node i as well as node j; as well as Ii is output from node i of sender layer. Output from a node j is computed thus:

$$O_j = f(net_j)$$

Function f represents a nonlinear sigmoidal function. Hidden layer node numbers are computed thus:

$$N_h = INT\sqrt{N_i \times N_0}$$

Wherein N_h represents quantity of hidden layer nodes, h represents hidden layer, N_i represents the quantity of input layer nodes, i represents quantity of input layer, N_o represents the quantity of output layer nodes, while o represents output layer. The equation proposed the quantity of hidden layer nodes as 6. However, after several tests, networks with 4 hidden layer nodes possessed the best outcome.

Activation function of unipolar sigmoid function is given below:

$$g(x) = \frac{1}{1+e^{-x}}$$

This is particularly beneficial in Neural Network trained through BP protocol.

Activation function of Bipolar sigmoid function is expressed through:

$$g(x) = \frac{1-e^{-x}}{1+e^{-x}}$$

This is similar to sigmoid functions.

Results and Discussion

For experiments, 5 lakh positive and 9 lakh negative twitter data was used. The features are selected using the proposed FA and compared to performance of the classifiers without feature selection. The experiments are run using the following combination: Without feature selection-NB, feature selection using FA-NB, without feature selection-KNN, feature selection using FA-KNN, without feature selection-MLPNN and feature selection using FA-MLPNN methods are evaluated. Figures 3-6 show the results of classification accuracy, Positive Predictive Value, Sensitivity and F Measure respectively.

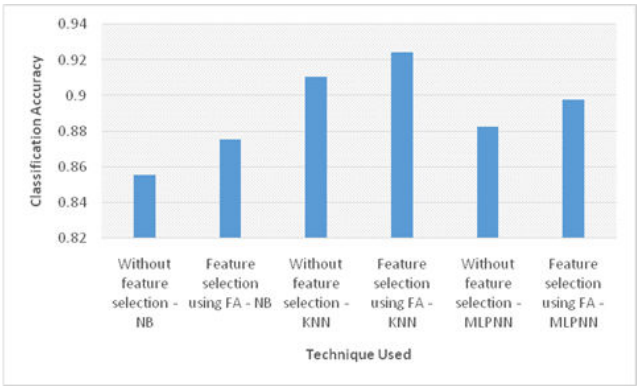

Figure 3. *Classification accuracy.*

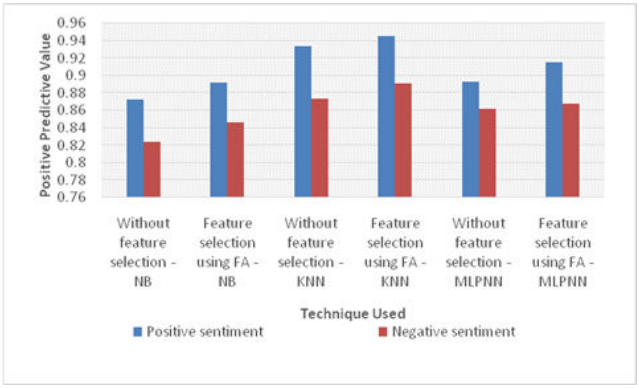

Figure 4. *Positive predictive value.*

From Figure 3 it is observed that the classification accuracy is improved for all classifiers with the proposed FA feature selection method. KNN classifier with proposed feature selection achieves the best classification accuracy of 92.42%.

Figure 4 shows that the average Positive Predictive Value of KNN with proposed FA is higher by 1.59% than without feature selection. Sensitivity is improved with the proposed FA feature selection as seen in Figure 5. The proposed method on an average improves sensitivity by 1.07%. The f measure of KNN with proposed FA feature selection performs better by 1.59% compared to no feature selection.

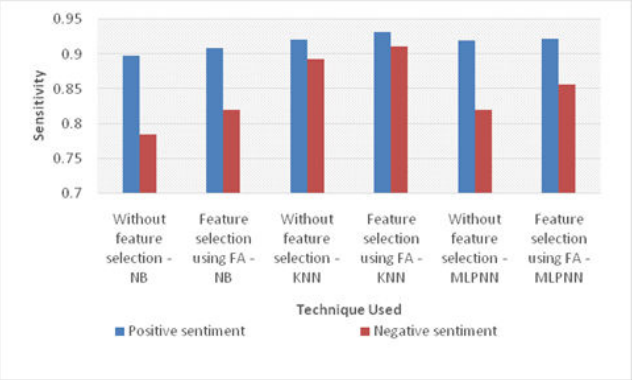

Figure 5. *Sensitivity.*

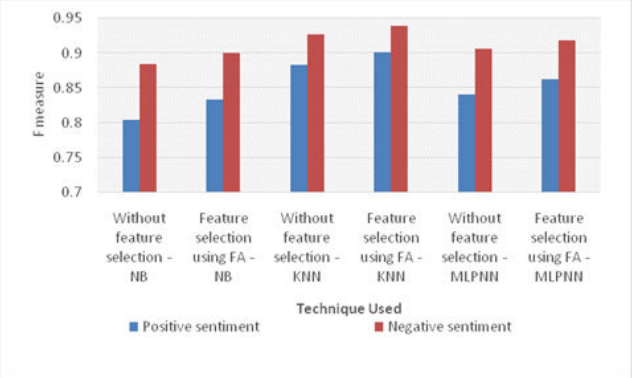

Figure 6. *F measure.*

Conclusion

The work provides basic concepts of Big Data. Extracting precious information from huge influx of data is crucial in Big Data analyses. Qualification as well as validation of every item in Big Data is not practical and therefore, novel methods are to be built. A firefly based feature selection is proposed to improve the classification of the twitter data. The feature selection not only reduces the dimensionality of the feature set but also reduces the time complexity. Outcomes of experiments prove that the suggested FA features selection technique enhances efficiency of classifiers. It is observed that the classification accuracy improves by 1.5% to 2.32%. Further investigation to optimize the classifiers is required.

References

1. Manyika J, Michael C, Brown B. Big data: The next frontier for innovation, competition, and productivity. Tech Rep, McKinsey, 2011.
2. Labrinidis A, Jagadish H. Challenges and opportunities with big data. Proceed VLDB Endowment 2012; 5: 2032-2033.
3. Han J, Kamber M, Pei J. Data Mining: Concepts and Techniques, Morgan Kaufmann, 2006.
4. Manyika J, Chui M, Brown B. Big data: the next frontier for innovation, competition, and productivity. McKinsey Global Institute, 2011.
5. Sri PA, Anusha M. Big Data-Survey. Indonesian J Elect Eng Informa (IJEEI) 2016.
6. Bello-Orgaz G, Jung JJ, Camacho D. Social big data: Recent achievements and new challenges. Information Fusion 2016; 28: 45-59.
7. Wu X, Zhu X, Wu GQ, Ding W. Data mining with big data. IEE Transact Knowledge Data Eng 2014; 26: 97-107.
8. Chen M, Mao S, Liu Y. Big data: A survey. Mobile Networks Appl 2014; 19: 171-209.
9. Chen J, Chen Y, Du X, Li C, Lu J, Zhao S, Zhou X. Big data challenge: a data management perspective. Frontiers Comput Sci 2013; 7: 157-164.
10. Junqué de Fortuny E, Martens D, Provost F. Predictive modeling with big data: is bigger really better?. Big Data 2013; 1: 215-226.
11. Gao W, Zhu Y, Jia Z, Luo C, Wang L, Li Z, Li X. Bigdatabench: a big data benchmark suite from web search engines, 2013.
12. Liu B, Blasch E, Chen Y, Shen D, Chen G. Scalable sentiment classification for big data analysis using Naive Bayes Classifier. In Big Data, 2013 IEEE International Conference, 2013.
13. Krishnamoorthy S, Saple AK, Achutharao PH. An integrated query optimization system for data grids. In Proceedings of the 1st Bangalore Annual Compute Conference. ACM, 2008.
14. Shafiq S, Butt WH, Qamar U. Attack type prediction using hybrid classifier. In Advanced Data Mining and Applications. Springer, Berlin, 2014.
15. Hu X, Weng Q. Estimating impervious surfaces from medium spatial resolution imagery using the self-organizing map and multi-layer perceptron neural networks. Remote Sensing Environ 2009; 113: 2089-2102.

***Correspondence to**

Senthamil Selvi R

MIET College

India