Analysis of Bayesian Optimization Algorithms for Big Data Classification Based on Map Reduce Framework

Chitrakant Banchhor (✉ banchhorchitrakant@gmail.com)

Dr. Vishwanathkarad MIT World peace university

Srinivasu N

Koneru Lakshmaiah Education Foundation

Research

Keywords: Map Reduce, correlative naive Bayes classifier, classification, big data, Holoentropy

Posted Date: December 23rd, 2020

DOI: <https://doi.org/10.21203/rs.3.rs-132776/v1>

License: © ⓘ This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

Analysis of Bayesian Optimization Algorithms for Big Data Classification Based on Map Reduce Framework

¹Chitrakant Banchhor and ²N.Srinivasu

¹School of Computer Engineering and Technology,

¹Dr. VishwanathKarad MIT World Peace University,

Mahatma Society, Kothrud, Pune, Maharashtra 411029, India

banchhorchitrakant@gmail.com

²Department of Computer Science and Engineering,

²Koneru Lakshmaiah Education Foundation,

Vaddeswaram, Andhra Pradesh 522502, India

Abstract. The process of big data handling refers the efficient management of storage and processing of very large volume of data. The data in a structured and an unstructured format require specific approach for overall handling. The classifiers analyzed in this paper are correlative naïve bayes classifier (CNB), Cuckoo Grey wolf CNB (CGCNB), Fuzzy CNB (FCNB), and Holoentropy CNB (HCNB). These classifiers are based on Bayesian principle and work accordingly. The CNB is developed by extending the standard naïve bayes classifier with applied correlation among the attributes so that it becomes a dependent hypothesis and it is named as a correlative naïve bayes classifier (CNB). The cuckoo search and grey wolf optimization algorithms are integrated with the CNB classifier and significant performance improvement is achieved. The resulting classifier is called as cuckoo grey wolf correlative naïve bayes classifier (CGCNB). The further performance improvements are achieved by incorporating fuzzy theory termed as fuzzy correlative naïve bayes classifier (FCNB) and holoentropy theory termed as Holoentropy correlative naïve bayes classifier

(HCNB) respectively. FCNB and HCNB classifiers are comparatively analyzed with CNB and CGCNB and achieved noticeable performance by analyzing with accuracy, sensitivity and specificity analysis.

Keywords: Map Reduce, correlative naive Bayes classifier, classification, big data, Holoentropy.

1. Introduction

The data size is increased because of the technological developments in the contemporary data computing field. The term big data refers the large size of dataset unable to manage and handle by the classical database systems [5].

The big data handling is a major challenge in every application due to the rapid growth in data gathering and storage, networking and other related techniques [6][58]. For example, big data classification certainly needed because of the huge capacity of data created as observed 2.5 quintillion bytes of data every day [7].

Big data refers the collection and processing of huge amount of data [8]. The definition of big data is based upon the facts, including its size, and inability of processing by certain systems due to high demand of primary storage and computing time [9][10].

The extensive voluminous data processed in a year leads to the frequent use of the term 'big data' in data analytics. This causes for the requirements that, every organization should have data management options available for handling of big data effectively [11][12].

The characterization of the big data is based on the factors such as volume, variety, and veracity associated with it. The enhanced difficulty level associated with the processing of big data is due to the fact, that data arrives in a continuous pattern from the internet sources [13][14].

Data analytics is considered as one of the challenging research problems and data mining and machine learning techniques are used to perform analytics. The inherent complexities associated with the data are its large size and non-uniform in nature makes the limitations on the current data mining software tools and techniques [15].

The standard data mining and machine learning methods are not directly dealing with extensive size data effectively [16][17]. The analysis and knowledge extraction process related to big data can be further improved.

The two major categories, namely clustering and classification included in data mining schemes. The big data classification process is primarily [18][19] influenced by the various classifiers, such as Naïve Bayes [20], Support Vector machine [21], and Extreme Learning Machine [22]. The data classification approach provides by ELM [23] algorithm is based on multiclassification rather than the binary classification [24]. The observed facts about big data processing is the increased computational complexity of because of the high volume data [25]. The analysis of big data in supervised classification is based on the learning algorithms, and thereafter, it finds the appropriate classes for the datasets[59].

The analytics include the extraction of useful insights from the extensive data and one of the major tasks associated with it is classification [26][27]. The deep study suggests for the redesign of typical classification algorithms so that it can be adopted in classification of very large amounts of data. The techniques required for classification of large amounts of data are considered as the crucial steps for different applications. The redesign of a classification technique for large amounts of data needs to be taken care, thereafter; the applications use these algorithms can have the limitations of the growth of the extensive size of data[28].

The analysis and organization techniques applicable for traditional systems are insufficient in addressing of big data related issues and challenges [29] [30]. There are a number of

algorithms developed for performing big data analysis tasks. The compatibility of MapReduce in handling big data processing makes it most suitable for analysis tasks [31].

The problem of processing a large size data is solved through MapReduce scheme which contains Mapper and Reducer tasks to be performed in parallel on datasets [32].

The Map Reduce principle derived from divide and conquer strategy of problem solving, in which sub-data samples are created and handled independently by splitting the data samples [33][34]. The Map level divides the source data by producing dissimilar pairs of the key value. The efforts obtained from the key value in the map function are integrated by reduce functions at the Reduce level [p4-4].

The number of smart data analytics approach such as image processing, multi-temporal processing and automatic classification are increasingly in demand due to the big data processing [35]. The data mining approaches integrated with the recent growing technologies are used for performing big data analytics with reduced limitations and drawbacks [36]. The MapReduce technique and distributed file system introduced by Google provides a suitable environment for the processing of the large-scale datasets over a group of systems. The MapReduce framework is used to perform big data mining using a number of processors efficiently [37]. Generally, Hadoop, one of the systems provides a parallel programming environment, [38] used for execution of MapReduce framework.

The classification algorithms are employed to solve data mining issues in big data because of the fact that it is the gathering of data from different sources. The classification task is based on the building of a classifier model for resultant target class prediction of data items included in the dataset [39]. The emphasis on data classification motivates the adaptive use of techniques for classification in the big data analysis. The classification techniques which include, Bayes networks, genetic algorithms, genetic programming, decision trees [40] are adopted for performing classification of large data.

It is found in the literature that big data classification is performed using machine learning methods [41][42], optimization algorithms [43][44], Decision Tree [45] primarily along with some other approaches.

The inclusion of fuzzy theory with correlative naïve bayes classifier creates a model named as the Fuzzy Correlative Naive Bayes Classifier (FCNB). The FCNB is developed for big data classification because it is implemented using the MapReduce framework.

Further enhancement of the CNB classifier is done by using the holoentropy function and the resultant model is the novel classification method developed for big data classification using the MapReduce model and named as Holoentropy based Correlative Naive Bayes Classifier (HCNB).

This paper is organized into different sections: Section II is based on Literature review covers review about the number of classifiers techniques developed using the MapReduce framework to perform classification on big data. Section III describes the developed Bayesian classifiers namely Correlative Naïve Bayes, Cuckoo Search Grey Wolves Correlative naïve Bayes, Fuzzy Correlative Naïve Bayes and Holoentropy based correlative naïve Bayes classifiers. The analysis of the developed classifiers is presented with comparisons based on the obtained results of accuracy, sensitivity and specificity. The final section includes conclusion and future scope for improvements of the classifiers discussed in this analysis.

2. Literature Review

In this section various algorithms for performing big data classification by different researchers and their importance in terms of advantages and disadvantages is presented.

Shen and Kai Gao [46] created a technique to be adopted for the big data classification, and it works in the internet environment. The performance of this approach was stable and also it is able to reduce the computation complexity. The approach has the lower complexity as

compare to the traditional segmentation approaches, but one of the demerits of this approach is its strongest feature independence.

An algorithm designed by Simone Scardapane *et al.* [47] for big data classification known as Echo State Networks based on the utilization of the multipliers optimization procedure. The method performed the local exchanges among the nearby elements by the effective utilization of the multiplier optimization procedure without assistance of connected nodes. In addition to this the communicating nodes did not require to use training patterns. The synthetic datasets were used in experimentation and showed improvements in computation time and generalization accuracy metrics. The method had an advantage of no requirements of pilot symbols during communication, whereas the weights did consider error values were needed to aggregate was one of the drawbacks of this approach.

Jemal H. Abawajy *et al.* [48] implemented large, iterative multitier ensemble (LIME) classifiers and analyzed its suitability for big data classification. This classifier integrated varieties of meta-classifiers at lower levels and an iterative system developed at the higher level. The results in experimentation showed that the developed classifier has improved performance as compared with existing classifiers for big data. The computational cost was a major drawback of the developed ensemble classifier.

Xin *et al.* [49] integrated Map Reduce framework and extreme learning machine to develop a classifier worked in a distributed environment. In the initial stage of this classifier learning weights computed through multiplication of data matrices. Further stage proposed elastic ELM with Map Reduce technique. The positive side of this approach was the rapid training speed and negligible human intervene because of its capability of efficient learning using dynamic processing massive training dataset.

Victoria López *et al.* [50] proposed an algorithm based on fuzzy rule based classification named as Chi-FRBCS-BigDataCS for dealing with imbalanced big data. This algorithm dealt

with the uncertainty in big data, with the strength of learning in the minimal class. This algorithm performs the fuzzy logic processes along with MapReduce concept, for design processes using the cost – sensitive learning approach. The performance of this algorithm is significantly improved in terms of classification accuracy and computation time.

Reshma C. Bhagat and Sachin S. Patil [51] performed classification of imbalanced data and created a multi-classifier. At its first step they utilized the techniques for conversion of the original datasets in the small group containing binary classes. In the next step balanced classes were formed from imbalanced binary class using the SMOTE algorithm. The classification was performed in the final step by utilization of the Random Forest classifier. The oversampling issues occurred in this algorithm were handled using the MapReduce concept so that the developed method acquired the capability of handling large datasets. The fixing of dataset dependent oversampling rates was one of the problems with this algorithm.

Triguero *et al.* [52] used evolutionary under sampling method and developed a parallel model for handling big data and its related issues. Processes involved in the classification task were distributed to various cluster computing nodes due to the MapReduce technique for the effective accomplishment of classification work. In addition, they also introduced a windowing approach for handling of class imbalanced data and it also increases the under – sampling speed. The advantage of this model was the scalability achievement during experimental studies. The abrupt change in computation time based on the imbalance ratio was the considerable drawback of this method

AlessioBechini *et al.* [53] developed a classifier by combining map reduce concept with association rules and then named it as MapReduce-based Associative Classifier (MRAC). This classifier utilized distributed classification method with map reduce model of programming and association rules. The method uses the FP-Growth algorithm and mined Classification Association Rules (CARs) and handled a distributed rule pruning classifying

unlabeled patterns. The advantage of this method was related to speed up and scalability, but it failed when performed experiments on the large datasets.

Shichao Zhang *et al.* [54] utilized the combination of clustering and classification approaches of K-Nearest Neighbors (KNN) and k-means clustering. The k-means clustering were used for dividing the entire dataset into various parts and kNN classification was performed in each part of the cluster. The results of the experiments performed on medical image and big data showed between efficiency and accuracy. This approach had a drawback of selecting a value of k because the degradation of classification accuracy when the value of k was increased.

The nature inspired meta-heuristic algorithm developed by SeyedaliMirjalili *et al.* [55] which was based on the social and hunting behavior of grey wolves. The algorithm adopted in computer science for optimization process and named as Grey wolves Optimization algorithm (GWO). The leadership hierarchy and hunting technique of the grey wolves were observed and accordingly the GWO algorithm was framed. The GWO algorithm has superior performance improvements when compared with other algorithms. The algorithms compared with GWO were Evolution Strategy (ES), Evolutionary Programming (EP), Differential Evolution (DE), Gravitational Search Algorithm (GSA), and Particle Swarm Optimization (PSO).

Cuckoo search is another meta-heuristic algorithm developed by Xin-She Yang and Suash Deb [56] to provide an approach towards the solution of optimization problems. The algorithm combines the brood parasitic behavior of the cuckoo species [60] with the Levy flight behavior of some birds and fruit flies. The experimental results of this algorithm showed improved performance as compared with genetic algorithm and particle swarm optimization techniques.

Simon Fong *et al.* [57] developed a method for mining insights from streaming data while avoiding the computational challenges. The developed algorithm was based on lightweight feature selection approaches and used to avoid high dimensionality of the accepted data in data mining. This method was suitable for real – time applications and showed effective in terms of high accuracy, low latency and robustness for solving the big data problems.

3. Descriptions of Classification methods

3.1. Naïve Bayes classifier (NB)

The different classifiers based on the bayesian theorem are known as bayesian classifiers, for example Naïve Bayes is one of the bayesian classifiers. The Bayesian classification is based on the posterior probability calculation by assumed prior probabilities and the probability of different data object under given assumptions.

The naïve bayes classifier is based on the fact that each attribute of the object to be classifies is independent of each other. The naïve bayes classifier is based on the approach where the probability of each category is calculated and the object belongs to the category with the largest probability associated.

3.2. Correlative Naive Bayes Classifier (CNB)

One of the highly utilized classifier is naïve bayes classifier, the typical classifier is adopted with map reduce framework and used for big data classification [1]. At the initial phase of the training process the input data are arranged in different groups on the basis of number of classes.

$$CNB_{Q \times m} = \{ \mu_{Q \times m}, \sigma_{Q \times m}, R_{Q \times 1} \} \quad (1)$$

Where, $\mu_{Q \times m}$ is the mean to be calculated, $\sigma_{Q \times m}$ is specified as variance, $R_{Q \times 1}$ denotes correlation function and it is illustrated in vector form. Testing data result is represented using the following equation:

$$C = \arg \max_{q=1 \dots Q} [P(C_q) \times P(X | C_q)] \times R^q \quad (2)$$

The equation (2) is used to indicate that the highest posterior value is only selected as consequential class.

3.3. Cuckoo Grey Wolf optimization with correlative Naïve Bayes classifier (CGCNB)

The integrated CGWO and CNB classifier with MapReduce framework is named as CGCNB-MRP [2]. CGWO is the integration of cuckoo search algorithm and grey wolf Optimization (GWO). The block diagram of developed model for big data classification is depicted in Figure-1.

Fig. 1. Control and Data Flow of CGCNB-MRP classifier

3.4. Fuzzy correlative Naive Bayes classifier (FCNB)

One more extension of CNB classifier is inclusion of fuzzy theory and it is named as FCNB [3]. FCNB classifier is shown in the following equation:

$$\mu_q^s = \frac{|m_q^s|}{d} \quad (3)$$

The term μ_q^s represents membership degree of symbol s^{th} in the q^{th} element of training model. The whole incidence of s^{th} symbol in q^{th} element is represented by the term $|m_q^s|$ and d is utilized for representing data sample in attribute. Each data sample is classified into number of classes let it be called as K . It is represented as follows:

$$\mu_c^k = \frac{|m^k|}{d} \quad (4)$$

Here whole incidence of K^{th} class in ground truth information is represented by $|m^k|$. The outcome of FCNB classifier is represented as below:

$$\text{FCNB} = \{ \mu_q^k, \mu_c^k, C \} \quad (5)$$

In the testing stage of developed FCNB classifier, testing sample is classified into the appropriate classes by the use of posterior probability of naïve bayes, fuzzy membership degree and a correlation factor among attributes. The output of FCNB is expressed as below equation:

$$G = \arg \max_{k=1 \text{ to } K} P(g_k | X) * C^k \quad (6)$$

The term here represented as $P(g_k | X)$ denoted as posterior probability by using test data X for given class g_k . The expression C^k signifies correlation for class K .

3.5.Holoentropy using correlative naïve bayes classifier for a big data classification

(HCNB)

The new classification technique called as HCNB [4] is introduced through combining existing CNB classifier with holoentropy function. The handling is based on the holoentropy estimation for each attribute using the following formula:

$$H_v^b = F \times T(\text{ib}) \quad (7)$$

Here, F represents the weight function and $T(\text{ib})$ is the entropy. The formulae for the weight function and entropy is described in the following equations.

$$F = 2 \left[1 - \frac{1}{1 + \exp(-T(i_b))} \right] \quad (8)$$

$$T(i_b) = - \sum_{b=1}^{M(i_b)} P_b \times \log P_b \quad (9)$$

Here, $M(i_b)$ is the unique value of the attribute vector i_b . The training phase of the HCNB based on the training data samples produces the result in the following vector form:

$$\text{HCNB}_{a \times s} = \{ \mu_{a \times s}, \sigma^2_{a \times s}, C_{a \times 1}, H_{a \times s} \} \quad (10)$$

Here, $\mu_{a \times s}$ and $\sigma^2_{a \times s}$ represent the computed mean value and computed variance value between the attributes a and s , respectively. Also, the correlation is represented by $C_{a \times 1}$ and holoentropy function is represented using $H_{a \times s}$. The individual class is selected by estimating posterior probability independently during a testing phase, which can be represented by below expression:

$$P(Y|k_v) = \prod_{b=1}^s P(Y = y_b | k_v) \quad (11)$$

where, y_b illustrates y^{th} data of b^{th} element, and k_v indicates v^{th} class number. The block diagram of the HCNB classifier is depicted in Figure 2.

Fig.2. Block diagram of HCNB classifier

4. Results and Discussion

This section presents the evaluation results of the classifiers and detail comparative analysis is provided with the existing methods. The system requirements and implementation details are provided in the experiment setup.

4.1. Experimental Setup

The system configuration used for performing experiment contains Windows 10 OS running on Intel processor CPU 2.16 GHz with 2 GB RAM capacity. The methods included in the developed classifiers are implemented in JAVA programming language. The dataset utilized for the experimentation purpose is localization dataset.

The localization dataset is taken from UCI machine learning repository for experimentation [61]. The recorded activities of five people who were wearing tags, ankle left, ankle right, belt, and chest is collected in terms of localization dataset. The dataset contains total 164860 instances that include 8 attributes. Each instance in the dataset forms a localization data for tags and attributes are used to recognize them.

4.2. Metrics used for Performance Evaluation

There are three metrics such as accuracy, sensitivity, and specificity is used for evaluation of the performance of the classifiers. The degree of veracity is measured using accuracy metric which is defined as the proportion of true results. The sensitivity and specificity are referred as the proportion of correctly identified true positives and true negatives respectively.

$$Acc = \frac{T_N + T_P}{T_N + T_P + F_N + F_P} \quad (A)$$

$$Sens = \frac{T_P}{T_P + F_N} \quad (B)$$

$$Spec = \frac{T_N}{T_N + F_P} \quad (C)$$

where, T_N is True Negative, T_P is True Positive, F_N is False Negative, and F_P is False Positive.

4.3. Comparative analysis of Bayesian models

A comparative analysis is done for performance evaluation of the developed classifiers with the existing models based on the sensitivity, specificity and accuracy analysis on localization dataset.

In this paper the first model CNB is compared with the existing naïve bayes classifier and thereafter further analysis shows the significant performance improvement of CGCNB in comparisons with NB and CNB. The other two developed models for big data classification named as FCNB and HCNB are compared with Naïve Bayes [27], Correlative Naive Bayes (CNB) [24], Cuckoo Grey Wolf based CNB (CGCNB), and Fuzzy Naïve Bayes classifier (FNB) [27].

The naïve bayes classifier with the incorporation of the fuzzy theory is named as the Fuzzy Naïve Bayes classifier (FNB), the comparison of FCNB classifier shows enhanced performance achievement and adopted to perform the big data classification.

The holoentropy extension of the CNB classifier (HCNB) is created and compared with the existing models with the similar condition and parameters and observed improvements when its performance is assessed on localization dataset. The comparative study of the models is done for training sample data taken from 75% to 90% for number of number mappers as $M=2,3,4$, and 5 on localization dataset.

4.4. Performance evaluation of CNB and CGCNB

The developed classifiers CNB and CGCNB are evaluated based on sensitivity, accuracy and specificity analysis on localization dataset. The performance evaluation is presented in this

section. The performance evaluation process is carried out on different number of mapper processes and on training data. The number of mappers used is referred as $M = 2$ to 5 and the data sample for training varies from 75% to 90%. Table 1 shows the analysis of CNB and CGCNB classifiers based on localization data.

Table1. Analysis of CNB and CGCNB based on localization data

Training Percentage (%)	Number of Mappers (M)	Bayesian Classifier Model (CNB)			Bayesian Classifier Model (CGCNB)		
		Acc %	Sens %	Spec %	Acc %	Sens %	Spec %
75	2	77.1	81	73	80	83	76.2
	3	77	81.3	73.2	80	83.5	76.3
	4	77.5	81.2	73.4	80.5	83.9	76.5
	5	77.1	81	73.4	80.1	83.6	76
80	2	77.5	81.3	73.2	80.4	83.9	76.3
	3	77.3	81.5	73.6	80.3	83.6	76.3
	4	77.6	81.5	73.6	80.5	84	76.6
	5	77.3	81	73.6	80.2	83.9	76.2
85	2	77.7	81.4	73.5	80.8	84	76.5
	3	77.4	81.7	73.8	80.3	84	76.7
	4	77.9	81.7	73.7	80.7	84.2	76.9
	5	77.6	81.4	73.7	80.4	84	76.7
90	2	77.8	81.7	73.9	80.9	84	76.6
	3	77.9	81.9	73.8	80.3	84.1	76.7
	4	77.9	81.9	73.9	80.8	84.4	76.9
	5	77.8	81.5	73.7	80.6	84.5	76.9

Acc = Accuracy, Sens = Sensitivity, Specificity = spec

4.5.Comparative Analysis of CGCNB

The comparison of considered Bayesian models NB, CNB, GWOCNB and CGCNB carried out based on sensitivity, accuracy and specificity metrics using localization dataset as given in table 2.

Table2. Analysis based on localization data

<i>Train ing data (%)</i>	<i>Mappers (M)</i>	<i>NB</i>			<i>CNB</i>			<i>GWO+CNB</i>			<i>CGCNB</i>		
		<i>Acc</i>	<i>Sen</i>	<i>Spe</i>	<i>Acc</i>	<i>Sens</i>	<i>Spe</i>	<i>Acc</i>	<i>Sen</i>	<i>Spe</i>	<i>Acc</i>	<i>Sen</i>	<i>Spe</i>
75	2	76.4	80.4	72.4	77.1	81	73	79.1	82.5	75.1	80	83.7	76.2
	3	76.1	80.5	72	77	81.3	73.2	79.4	82.9	75.3	80	83.5	76.3
	4	76.1	80.2	72.2	77.5	81.2	73.4	79.2	82.9	75.2	80.5	83.5	76.5
	5	76.3	80.1	72.1	77.1	81	73.4	79.2	82.9	75.1	80.1	83.6	76.2
80	2	76.7	80.6	72.5	77.5	81.3	73.2	79.2	82.7	75.3	80.4	83.5	76.3
	3	76.5	80.5	72.1	77.3	81.5	73.6	79.5	82.5	75.5	80.3	83.6	76.3
	4	76.2	80.5	72.5	77.6	81.5	73.6	79.5	83	75.3	80.5	84	76.6
	5	76.5	80.3	72.1	77.3	81	73.6	79.5	82.9	75.1	80.2	83.5	76.2
85	2	76.8	80.7	72.7	77.7	81.4	73.5	79.4	82.9	75.4	80.5	84	76.5
	3	76.7	80.6	72.5	77.4	81.7	73.8	79.6	83	75.7	80.3	84	76.7
	4	76.2	80.7	72.5	77.9	81.7	73.7	79.7	83.2	75.7	80.7	84.2	76.9
	5	76.9	80.4	72.7	77.6	81.4	73.7	79.6	83	75.2	80.4	84	76.7
90	2	76.9	80.9	72.9	77.8	81.7	73.9	79.9	82.9	75.7	80.9	84	76.6
	3	76.8	80.8	72.8	77.9	81.9	73.8	79.7	83.3	75.7	80.4	84.1	76.7
	4	76.5	80.7	72.9	7.9	81.9	73.9	79.9	83.2	75.9	80.9	84.4	76.9
	5	76.9	80.9	72.7	7.8	81.5	73.7	79.8	83.4	75.9	80.7	84.5	76.9

Analyze the model with 75% of training data

The CGCNB model observed sensitivity, specificity and accuracy analysis as 80%, 83.7% and 76.2% respectively as compared to NB classifier's attained analysis of these three metrics as 76.4%, 80.4%, and 72.4% for mappers 2, 3, 4 and 5 at 75% of training based on localization data.

The NB model has attained 76.4% accuracy, 80.4% sensitivity and 72.4% specificity with 75% training data using localization dataset for M=2, 3, 4 and 5. The CGCNB has analysis values on similar conditions as 80%, 83.7%, and 76.2% for accuracy, sensitivity and specificity respectively.

The CGCNB has the accuracy 80.1%, which is superior to the GWO+CNB of 79.2% for mapper 5. The analysis showed that the CGCNB obtained maximum accuracy, sensitivity and specificity as 80.5%, 83.9%, and 76.5% with 75% training when performed on 4 mappers.

Analyze the model with 80% of training data

The maximum performance achieved is 80.5% for accuracy, 84% for sensitivity, and 76.6% for specificity with 80% training by 4 mappers using localization data. On the other side the Gary wolf optimization with CNB have accuracy value 79.5%, sensitivity value 83%, and specificity value 75.3%.

Analyze the model when apply Maximum percentage of training sample data

The performance of CGCNB method when maximum training sample data and 5 mappers are used is 80.7%, 84.5%, and 76.9% of accuracy, sensitivity, and specificity values respectively. This is far better than the performance obtained by other Bayesian methods based on naïve bayes approach. The accuracy and sensitivity obtained by existing classifiers is 76.9%, 80.9% for NB, 77.8%, 81.5% for CNB, and 79.8%, 83.4% for GWO+CNB. The specificity of CGCNB is also superior that the existing method.

The CGCNB classifier has enhanced performance based on three metrics as compare to other methods, namely NB, CNB and GWO+CNB.

4.6. Comparative Analysis of FCNB

The FCNB model is compared with other methods including CNB and Fuzzy Naïve Bayes (FNB) on the basis of sensitivity, specificity and accuracy analysis. The varying training percentage from 75% to 90% with different mapper size as 2, 3, 4 and 5 using localization dataset were observed.

Accuracy analysis of classifiers

The Figure 3 (a) illustrates the accuracy analysis of FCNB with other models. The accuracy value observed by FCNB on 90% training with mapper size as 4 is 91.7816%, which is the significant improvement of the accuracy of existing classifiers on the same parameters. The existing classifiers attained accuracy values as 76.504% for NB, 77.9505% for CNB, 79.862% for GWO+CNB, 80.8977% for CGCNB and 72.33% for FNB.

The comparison based on accuracy analysis is done when the number of mappers is 5 and training data is 90% using localization dataset and is illustrated in Figure 3 (b).

The achieved accuracy value is 76.996% for NB, 77.808% for CNB, 79.7638% for GWO+CNB, 80.6568% for CGCNB, and 72.3310% for FNB. In comparison with the existing classifiers the high accuracy value 91.781664% is observed for FCNB.

Figure3. FCNB Model comparison on accuracy analysis for M= 4 and M=5 using localization Dataset

Sensitivity analysis of classifiers

Figure 4 illustrates the comparative sensitivity analysis of FCNB classifier with other existing classifiers with variations on training percentage and number of mappers using localization dataset.

Figure 4 (a) presents the sensitivity analysis when 90% of training data applied with the number of mappers as 4 using localization dataset. The sensitivity values obtained 80.699% for NB, 81.899% for CNB, 83.2474% for GWO+CNB, 84.399% for CGCNB, 99.986% for FNB, and 94.79% for FCNB. The sensitivity value of FCNB is less as compare to FNB because of the classification based on the selected training data classified towards the same class.

The sensitivity performance with 90% training with 5 mappers is presented in Figure 4(b). The sensitivity value obtained by the existing classifiers is 80.879% for NB, 81.4741% for CNB, 83.447% for GWO+CNB, 84.478% for CGCNB, and 99.98% for FNB. In comparison with this FCNB has achieved sensitivity value of 94.79%.

Figure4. FCNB Model comparison on sensitivity analysis for M= 4 and M=5 using localization Dataset

Specificity analysis of classifiers

The specificity analysis of the FCNB classifier with varying mappers and percentage of training using the localization dataset is compared and illustrated in Figure 5.

Figure 5 (a) shows the specificity performance of classifiers when the 4 mappers are used and training data is 90%. The specificity values achieved by existing classifiers are 72.884%,

73.953%, 75.9060%, 76.9617%, and 36.12%, respectively, whereas FCNB model has specificity value of 88.891% which is considered as a significant improvement.

Figure 5 (b) illustrates similar specificity analysis with 5 mappers for 90% training of localization data and the obtained values of NB, CNB, GWO+CNB, CGCNB and FNB are 72.7007%, 73.8631%, 75.8758%, 76.99%1, and 36.127%, respectively. For the same condition the FCNB classifier observed high specificity value of 88.89%.

Figure5. FCNB Model comparison on specificity analysis for M= 4 and M=5 using localization Dataset

4.7. Comparative Analysis of HCNB

The HCNB classifier is compared with other considered classifiers. The three metrics analysis of HCNB is compares with other Bayesian models which includes GWO+CNB, and CGCNB also. The analysis is performed by varying training and mapper sizes using the localization data. The HCNB classifier has improved performance based on the analysis illustrated in this section.

Accuracy analysis of classifiers

The accuracy analysis is illustrated in Figure 6, when the mapper size is 4 for the localization dataset. The percentage of accuracy obtained from the existing models is 76.1438% for NB, 77.4994% for CNB, 79.1825% for GWO+CNB, and 80.5052% for CGCNB, whereas the HCNB has attained accuracy 93.5965%. The HCNB model has better accuracy as compared with other models and it is shown in the Figure 6 (a) using localization dataset.

The similar observation is described in Figure 6(b), with 5 mappers on 75% of training data using the localization dataset. The attained accuracy value for the classifiers is 76.2616% for NB, 77.0957% for CNB, 79.1593% for GWO+CNB, and 80.1475% for CGCNB. In this case also HCNB model has enhanced accuracy, observed is 93.5965%.

Figure6. HCNB Model comparison on accuracy analysis for M= 4 and M=5 using localization Dataset

Sensitivity analysis of classifiers

Figure 7 (a) illustrates, the observed sensitivity values of classifiers are compared, when the number of mappers M is 4 and training data taken is 80% using localization dataset.

The sensitivity values obtained from different classifier models is 80.4893% for NB, 81.5303% for CNB, 83.0372% for GWO+CNB, and 84.0364% for CGCNB, whereas it is observed that HCNB obtained improved sensitivity value as 94.336%.

In the similar note the sensitivity analysis of the classifiers is done with 5 mappers and the result is depicted in Figure 7(b). The observed sensitivity value is 80.3045% for NB, 81.0207% for CNB, 82.9549% for GWO+CNB, 83.8788% for CGCNB, and 94.336 for HCNB.

Figure7. HCNB Model comparison on sensitivity analysis for M= 4 and M=5 using localization Dataset

Specificity analysis of classifiers

Figure 8(a) shows the specificity results of different classifiers based on 4 mappers with 75% training using localization dataset. The specificity values obtained in terms of percentages is 72.2145% for NB, 73.4195% for CNB, 75.2482% for GWO+CNB, 76.5403% for CGCNB, and 84% for HCNB.

The following Figure 8(b) shows the comparative analysis based on specificity with mapper size M = 5, on 75% training using localization data and obtained results as 72.1371% for NB, 73.4455% for CNB, 75.1138% for GWO+CNB, 76.0083% for CGCNB, and 84% for HDNB.

Figure8. HCNB Model comparison on specificity analysis for M= 4 and M=5 using localization Dataset

5. Conclusion

This paper focused on big data classification based on different functions incorporated with mapreduce framework. The basic model is CNB classifier and later it is integrated with optimization algorithms, like cuckoo search and grey wolf optimization. The adoption of fuzzy theory with CNB classifier with membership degree of attributes included in the dataset provides performance achievements comparatively with CNB and CGCNB classifiers. From simulation outcomes, the models, such as FCNB, HCNB and CGCNB

classifier demonstrates the enhanced performance in localization and a skin segmentation databases. In future, the performance of the classifiers will be analyzed using log loss and training loss.

Declarations

Availability of data and materials: The data used for the analysis is taken from the localization dataset, available at, <https://archive.ics.uci.edu/ml/datasets/Localization+Data+for+Person+Activity>

Competing interests: None

Funding: None

Authors' contributions: All authors contributed to the design and implementation of the research, to the analysis of the results and to the writing of the manuscript.

Acknowledgements: I would like to express my very great appreciation to Dr. N. Srinivasu for his valuable and constructive suggestions during the planning and development of this research work. His willingness to give his time so generously has been very much appreciated. Also, I wish to thank my parents for their support and encouragement throughout my study.

References

- [1].Chitrakant Banchhor, N. Srinivasu (2016) CNB-MRF: Adapting Correlative Naive Bayes Classifier and MapReduce Framework for Big Data Classification. International Review on Computers and Software (IRECOS). <https://doi.org/10.15866/irecos.v11i11.10116>
- [2].Chitrakant Banchhor, N. Srinivasu (2020) Integrating Cuckoo search-Grey wolf optimization and Correlative Naive Bayes classifier with Map Reduce model for big data

classification. Data and Knowledge Engineering.

<https://doi.org/10.1016/j.datak.2019.101788>

- [3].ChitrakantBanchhor, N. Srinivasu (2018) FCNB: Fuzzy Correlative Naive Bayes Classifier with MapReduce Framework for Big Data classification. Journal of intelligent systems. <https://doi.org/10.1515/jisys-2018-0020>
- [4].ChitrakantBanchhor, N. Srinivasu (2019) Holoentropy based Correlative Naive Bayes classifier and MapReduce model for classifying the big data. Evolutionary Intelligence. <https://doi.org/10.1007/s12065-019-00276-9>
- [5].Sidahmed Benabderrahmane, Nedra Mellouli, Myriam Lamolle, and Patrick Paroubek, "Smart4Job: A Big Data Framework for Intelligent Job Offers Broadcasting Using Time Series Forecasting and Semantic Classification," Big Data Research, vol. 7, pp. 16-30, 2017.
- [6].Thanekar, S.A., Subrahmanyam, K., & Bagwan, A. B. (2016).Big data and MapReduce challenges, opportunities and trends.International Journal of Electrical and Computer Engineering, 6(6), 2911-2919. doi:10.11591/ijece.v6i6.10555
- [7].Wu, X., Zhu, X., Wu, G.Q., Ding, W., "Data mining with big data", IEEE Transactions on Knowledge and Data Engineering, vol. 26, no. 1, pp. 97–107, 2014.
- [8].V. Marx, "The big challenges of big data", Nature, vol. 498, no. 7453, pp. 255–260, 2013.
- [9].M. Minelli, M. Chambers, A. Dhiraj, "Big Data, Big Analytics: Emerging Business Intelligence and Analytic Trends for Today'sBusinesses", 1st Edition, Wiley Publishing, 2013.
- [10]. Pole, G., & Gera, P. (2016). A recent study of emerging tools and technologies boosting big data analytics doi:10.1007/978-981-10-0419-3_4
- [11]. W. Lin, Z. Wu, L. Lin, A. Wen and J. Li, "An Ensemble Random Forest Algorithm for Insurance Big Data Analysis," IEEE Access, vol. 5, pp. 16568-16575, 2017.

- [12]. S.S. Patil and S.P. Sonavane, "Enriched Over_Sampling Techniques for Improving Classification of Imbalanced Big Data," In Proceedings of IEEE Third International Conference on Big Data Computing Service and Applications (Big Data Service), San Francisco, CA, pp. 1-10, 2017.
- [13]. J. Chen, H. Chen, X. Wan and G. Zheng, "MR-ELM: a MapReduce-based framework for large-scale ELM training in big data era," Neural Computing and Applications, vol. 27, no 1, pp.101-110, 2016.
- [14]. Radha, K., & Thirumala Rao, B. (2016). Research issues and challenges of big data. International Journal of Control Theory and Applications, 9(17), 8437-8444.
- [15]. Chih-Fong Tsai, Wei-Chao Lin, Shih-Wen Ke, "Big data mining with parallel computing: A comparison of distributed and MapReduce methodologies", Journal of Systems and Software, Vol. 122, pp. 83–92, December 2016.
- [16]. Victoria López, Sara del Río, José Manuel Benítez, Francisco Herrera, "Cost-sensitive linguistic fuzzy rule based classification systems under the MapReduce framework for imbalanced big data", Fuzzy Sets and Systems, Vol. 258, pp. 5–38, January 2015.
- [17]. Sucharita, V., Jyothi, S., & Rao, P. V. (2016). Comparison of machine learning algorithms for classification of penaeid prawn species. Paper presented at the Proceedings of the 10th INDIACom; 2016 3rd International Conference on Computing for Sustainable Global Development, INDIACom 2016, 1610-1613.
- [18]. A. Haque, B. Parker, L. Khan and B. Thiraisingham, "Evolving Big Data Stream Classification with MapReduce," In Proceedings of IEEE 7th International Conference on Cloud Computing, Anchorage, AK, pp. 570-577, 2014.
- [19]. O. Hegazy, S. Safwat, and M.E. Bakry, "A mapreduce fuzzy techniques of big data classification," In Proceedings of the SAI Computing Conference (SAI), London, pp. 118-128, 2016.

- [20]. G. Santafe, J.A. Lozano and P. Larranaga, "Bayesian Model Averaging of Naive Bayes for Clustering," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 36, no. 5, pp. 1149-1161, October 2006.
- [21]. X. Huang, L. Shi and J.A.K. Suykens, "Support Vector Machine Classifier With Pinball Loss," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 5, pp. 984-997, May 2014.
- [22]. G.B. Huang, Q.Y. Zhu, and C.K. Siew, "Extreme learning machine: theory and applications," *Neurocomputing*, vol. 70, no. 1, pp.489-501, 2006.
- [23]. J. Chen, H. Chen, X. Wan and G. Zheng, "MR-ELM: a MapReduce-based framework for large-scale ELM training in big data era," *Neural Computing and Applications*, vol. 27, no 1, pp.101-110, 2016.
- [24]. M. Duan, K. Li, X. Liao and K. Li, "A Parallel Multiclassification Algorithm for Big Data Using an Extreme Learning Machine," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, no. 6, pp. 2337 - 2351, 2017.
- [25]. Á. Arnaiz-González, A.González-Rogel, J.F. Díez-Pastorand C.López-Nozal, "MR-DIS: democratic instance selection for big data by MapReduce," *Progress in Artificial Intelligence*, vol. 6, no. 3, pp. 211–219, September 2017.
- [26]. A. Segatori, F. Marcelloni and W. Pedrycz, "On Distributed Fuzzy Decision Trees for Big Data," *IEEE Transactions on Fuzzy Systems*, vol. 26, no. 1, pp. 174-192, 2018.
- [27]. A.Bechini, F. Marcelloni, and A. Segatori, "A MapReduce solution for associative classification of big data," *Information Sciences*, vol. 332, pp.33-55, 2016.
- [28]. Manekar, A.K., & Pradeepini, G. (2016). Cloud based big data analytics a review. Paper presented at the Proceedings - 2015 International Conference on Computational Intelligence and Communication Networks, CICN 2015, 785-788. doi:10.1109/CICN.2015.160

- [29]. H. Hu, Y. Wen, T. S. Chua, and X. Li, "Toward scalable systems for big data analytics: A technology tutorial," *IEEE Access*, vol. 2, pp. 652–687, 2014.
- [30]. Alessio Bechini, Francesco Marcelloni, and Armando Segatori, "A MapReduce solution for associative classification of big data," *Information Sciences*, vol. 332, pp. 33-55, 2016.
- [31]. Anushree Priyadarshini, and Sonali Agarwal, "A map-reduce based support vector machine for big data classification," *International Journal of Database Theory and Application*, vol. 8, no .5, pp.77-98, 2015.
- [32]. R. Bhukya and J. Gyani, "Fuzzy associative classification algorithm based on MapReduce framework," In *Proceedings of the International Conference on Applied and Theoretical Computing and Communication Technology (iCATccT)*, Davangere, pp. 357-360, 2015.
- [33]. Jeffrey Dean, and Sanjay Ghemawat, "MapReduce: Simplified data processing on large clusters," *ACM Communications*, vol. 51, no. 1, pp. 107–113, 2008.
- [34]. Mikel Elcano, Mikel Galar, Jose Sanz, and Humberto Bustince, "CHI-BD: A Fuzzy Rule-Based Classification System for Big Data classification problems," *Fuzzy Sets and Systems*, pp. 1-38, 2017.
- [35]. G. Cavallaro, M. Riedel, M. Richerzhagen, J. A. Benediktsson and A. Plaza, "On Understanding Big Data Impacts in Remotely Sensed Image Classification Using Support Vector Machine Methods," in *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 8, no. 10, pp. 4634-4646, Oct. 2015.
- [36]. Isaac Triguero, Daniel Peralta, Jaume Bacardit, Salvador García, Francisco Herrera, "MRPR: A MapReduce solution for prototype reduction in big data classification", *Neurocomputing*, Vol. 150, Part A, pp. 331–345, February 2015.

- [37]. Chih-Fong Tsai, Wei-Chao Lin, Shih-Wen Ke, "Big data mining with parallel computing: A comparison of distributed and MapReduce methodologies", *Journal of Systems and Software*, Vol. 122, pp. 83–92, December 2016.
- [38]. Dean, J. and Ghemawat, S., "Map reduce: a flexible data processing tool", *Communications of the ACM*, vol. 53, no. 1, pp. 72-77, 2010.
- [39]. M.S. Kamal, S. Parvin, A.S. Ashour, F. Shiand N. Dey, "De-Bruijn graph with MapReduce framework towards metagenomic data classification," *International Journal of Information Technology*, vol. 9, no. 1, pp.59-75, 2017.
- [40]. Á.Arnaiz-González, A.González-Rogel, J.F. Díez-Pastorand C.López-Nozal, "MR-DIS: democratic instance selection for big data by MapReduce," *Progress in Artificial Intelligence*, vol. 6, no. 3, pp. 211–219, September 2017.
- [41]. Avinash Ratre, "Taylor series based compressive approach and Firefly support vector neural network for tracking and anomaly detection in crowded videos," *Journal of Engineering Research*, vol. 7, no. 4, 2019.
- [42]. Arul. V.H, V.G. Sivakumar, Ramalatha Marimuthu, and Basabi Chakraborty, "An Approach for Speech Enhancement Using Deep Convolutional Neural Network," *Multimedia Research*, vol. 2, no. 1, pp. 37-44, 2019.
- [43]. Rajesh B. Ingle and Nitin S. More, "Energy-aware VM migration using Dragonfly-Crow optimization and support vector regression model in Cloud Data," *International Journal of Modeling, Simulation, and Scientific Computing*, 2018.
- [44]. Vijayakumar Polepally and K Shahu Chatrapati, "Dragonfly optimization and constraint measure-based load balancing in cloud computing," *Cluster Computing*, pp. 1-13, 2107.
- [45]. B. S. Daga and Avinash N. Bhute, "Predicting recurrence pattern in breast cancer Using Decision Tree," 2009.

- [46]. ShenGao, Kai Gao, "Modelling on Classification and Retrieval Strategy in Map-Reduce Based IR System", in proceedings of 2014 International Conference on Modelling, Identification and Control, Melbourne, Australia, December 3-5, 2014.
- [47]. Simone Scardapane, Dianhui Wang, Massimo Panella, "A Decentralized Training Algorithm for Echo State Networks in Distributed Big Data Applications", Neural Networks, August 2015.
- [48]. Jemal H. Abawajy, Andrei Kelarev, and Morshed Chowdhury, "Large Iterative Multitier Ensemble Classifiers for Security of BigData", IEEE Transactions on Emerging Topics in Computing, vol. 2, no. 3, pp.352 - 363, April 2014.
- [49]. Junchang Xin, Zhiqiong Wang, Luxuan Qu, Guoren Wang, "Elastic extreme learning machine for big data classification", Neurocomputing, vol. 149, pp. 464–471, 2015.
- [50]. Victoria López, Sara del Río, José Manuel Benítez, Francisco Herrera, "Cost-sensitive linguistic fuzzy rule based classification systems under the MapReduce framework for imbalanced big data", Fuzzy Sets and Systems, Vol. 258, pp. 5–38, January 2015.
- [51]. Reshma C. Bhagat, Sachin S. Patil, "Enhanced SMOTE Algorithm for Classification of Imbalanced Big-Data using Random Forest", in proceedings of IEEE International on Advance Computing Conference (IACC), pp. 403 - 408, 2015.
- [52]. I. Triguero, M. Galar, S. Vluymans, C. Cornelis, H. Bustince, F. Herrera and Y. Saeys, "Evolutionary Undersampling for Imbalanced Big Data Classification", Evolutionary Computation, Vol. 17, No. 3, pp. 275-306, 2009.
- [53]. Alessio Bechini, Francesco Marcelloni, Armando Segatori, "A MapReduce solution for associative classification of big data", Information Sciences, Vol. 332, pp. 33–55, March 2016.

- [54]. Zhenyun Deng, Xiaoshu Zhu, Debo Cheng, Ming Zong, Shichao Zhang, "Efficient kNN classification algorithm for big data", *Neurocomputing*, Vol. 195, pp. 143–148, June 2016.
- [55]. SeyedaliMirjalili, Seyed Mohammad Mirjalili, Andrew Lewis, "Grey Wolf Optimizer", *Advances in Engineering Software*, Vol. 69, pp. 46–61, March 2014.
- [56]. X.-S. Yang, S. Deb, "Cuckoo search via Levy flights", in *Proceedings of World Congress on Nature \& Biologically Inspired Computing (NaBIC 2009)*, Coimbatore, IEEE Publications, USA, pp. 210-214. December 2009.
- [57]. Simon Fong, Raymond Wong, and Athanasios V. Vasilakos, "Accelerated PSO swarm search feature selection for data stream mining big data," *IEEE transactions on services computing*, vol. 9, no. 1, pp. 33-45, 2016.
- [58]. Raghav, R. S.; Amudhavel, J.; Dhavachelvan, P., "A SURVEY ON TOOLS USED IN BIG DATA PLATFORM", *Advances and Applications in Mathematical Sciences* Volume 17, Issue 1, November 2017, Pages 213-229, Mili Publications
- [59]. Potharaju, Sai Prasad; Sreedevi, M., "Distributed feature selection (DFS) strategy for microarray gene expression data to improve the classification performance", *Clinical Epidemiology And Global Health*, Volume 7, Issue 2, June 2019, Pages 171-176
- [60]. Thirugnanasambandam, K., Prakash, S., Subramanian, V. et al. Reinforced cuckoo search algorithm-based multimodal optimization. *ApplIntell* 49, 2059–2083 (2019).<https://doi.org/10.1007/s10489-018-1355-3>
- [61]. Localization dataset, available at, <https://archive.ics.uci.edu/ml/datasets/Localization+Data+for+Person+Activity>

Figures

Figure 1

Control and Data Flow of CGCNB-MRP classifier

Figure 2

Block diagram of HCNB classifier

(a)

(b)

Figure 3

FCNB Model comparison on accuracy analysis for M= 4 and M=5 using localization Dataset

(a)

(b)

Figure 4

FCNB Model comparison on sensitivity analysis for M= 4 and M=5 using localization Dataset

(a)

(b)

Figure 5

FCNB Model comparison on specificity analysis for M= 4 and M=5 using localization Dataset

Figure 6

HCNB Model comparison on accuracy analysis for M= 4 and M=5 using localization Dataset

Figure 7

HCNB Model comparison on sensitivity analysis for M= 4 and M=5 using localization Dataset

(a)

(b)

Figure 8

HCNB Model comparison on specificity analysis for M= 4 and M=5 using localization Dataset