Decision Stump Feature Selection Based Mean Shift Brown Boost Map Reduce Clustering For Predictive Analytics With Big Data

Mrs. Anita M[1], Dr. Shakila S[2]
***Corresponding Author**:- Mrs. Anita M
[1*,2]The Governement Arts College, Department of Computer Science, Trichy-620022, Tamilnadu.,E-mail:[1]anitarajkumar040908@gmail.com , [2]shakilamuthusamy@gmail.com

Abstract

Big data refers to the generation of a huge volume of data continuously. Hence, analytics on such as large volume of data is becoming more complex regarding more time consumption and memory usage. With the aim of enhancing prediction accuracy by lesser time consumption, Decision Stump Feature Selection based Mean Shift Brown Boost Map Reduce Data Clustering (DSFS-MSBBMPDC) Technique is introduced for analyzing the spatial data to predict the future results. DSFS-MSBBMPDC technique consists of various procedures such as feature selection and clustering process to predict future results. First, the Otsuka-Ochiai decision stump Feature Selection was performed for choosing significant features. By one internal node, decision tree is linked to terminal node. After feature selection, the mean Shift steepest descent Brown Boost Map Reduce Data clustering process was performed to group input data to perform spatial data analysis. Brown Boost cluster combines the weak learner to form strong cluster. The prediction accuracy was increased as well as prediction error was reduced using the steepest descent function. The simulation is achieved by geographical dataset with various parameters by amount of features and amount of data. DSFS-MSBBMPDC improves performance compared with state-of-the-art works.

Keywords: Predictive Analytics, big data, Otsuka-Ochiai decision stump, Brown Boost cluster, Mean Shift Clustering, steepest descent function

DOI Number: 10.14704/nq.2022.20.9.NQ44665 **Neuro Quantology 2022; 20(9):5698-5706**

1. Introduction

By current expansion as well as innovation in IT, data is formed by extremely higher rate into different fields. Therefore, novel technology has needed for storing and extracting valuable information but the detection and removal of instruction and awareness via data volume were tricky. Data clustering is the procedure of partitioning numerous data within distinct subsets by means of certain similarity/dissimilarity measures.

Map Reduce implementations of the Hierarchical Density-Based Clustering (MR-HDBSCAN*) were introduced in [1] with big data. But, the computational time of the Density-Based Clustering process was not minimized. An intelligent weighting *k* -means clustering (IWKM) algorithm was introduced in [2]. But it Failed to apply the high-dimensional data with dissimilar big data applications.

Transductive LSTM (T-LSTM) was introduced in [3] for time-series prediction to estimate dissimilar weather environment. However, improved prediction accuracy was not obtained. An incremental K-means clustering is developed in [4] to increase accuracy and minimize time. Designed algorithm was unsuccessful to reduce space complexity.

A k-bat clustering algorithm was introduced in [5] to assess the big scale datasets. However, the designed algorithm failed to apply into a real-world datasets of different fields. A parallel batch clustering is designed in [6] depend on k-means algorithm. Designed method was failed to enhance accuracy.

Weighted consensus clustering technique was Developed in [7] to achieve the highest accuracy based on squared Euclidean distance. It was tricky for creating final decision. A firefly method is developed in [8] for identifying perfect clustering by greater accuracy. However, space complexity analysis is not achieved to improve accuracy.

A complex hierarchical clustering is designed in [9] for minimizing time. But, designed method was unsuccessful for improving clustering performances. A new hybrid clustering is designed in [10] for increasing precision and recall. Designed method was unsuccessful for decreasing execution time.

Density-based clustering approach was developed in [11] for improving the clustering quality. However, the complexity analysis of clustering was not performed. A novel approach called spectral clustering was designed in [12] for distributed data, with minimum communication overhead. But, the higher accuracy for distributed data clustering was not achieved.

A Distributed Clustering Algorithm (DCA) was introduced in [13] based on big data processing. However, the complexity of the algorithm was higher. A hyper plane division method was introduced in [14] to increase the clustering procedure with efficient, and effective. However, the division method failed to focus on big data with large data and high feature dimensionality.

A novel distributed network CASS was developed in [15] with minimum running time. But the algorithm was not efficient for increasing performance of clustering for huge-scale dataset. An efficient and scalable parallel varied density-based clustering approach is developed in [16] for improving performance as well as minimizes time. But, it was unsuccessful for improving efficiency of the clustering approach for the real-time data set.

Map Reduce-based priori algorithm was designed in [17] for Big Data analytics by lesser time. But, designed method was unsuccessful for enhancing analysis of large clusters. Distributed Fuzzy Neural Network (DFNN) was developed in [18] for big data analytics. But the multiple heterogeneous features of big data were not considered.

Novel intra cluster cohesion was introduced in [19] based on the data similarity within a cluster. But the computational complexity of clustering was not minimized while handling the large dataset. A correlation-based clustering technique was developed in [20] with big patio-temporal datasets. However, the forecast errors over space and time were not minimized.

To overcome the issues, a technique called DSFS-MSBBMPDC is introduced and the novel contributions are summarized as follows,

❖ To enhance prediction, DSFS-MSBBMPDC is introduced with big data based on feature selection and ensemble clustering.

❖ The DSFS-MSBBMPDC uses the Otsuka-Ochiai decision stump-based feature selection to discover the significant features based on similarity measures. As a result, the significant features are selected for further processing as a result of prediction time and space complexity is minimized.

❖ To improve accuracy as well as reduce false positive rate, mean Shift steepest descent Brown Boost Map Reduce Data clustering was designed in DSFS-MSBBMPDC. Steepest descent function is employed for detecting the results with lesser error.

❖ Finally, comprehensive experiments assessments are carried out to estimate the performance of our DSFS-MSBBMPDC and other related works along with the various performance metrics.

The rest of this article is summarized by: Section 2 discussed briefly about DSFS-MSBBMPDC technique. Section 3 explains simulation settings of proposed and existing methods. Section 4 provides results and discussion. Section 5 describes conclusion of article.

2. Methodology

Big data analytics was a method of gathering, managing as well as analyzing enormous sets of data for establishing patterns or valuable instructions. In recent days, various research works is designed for big data analytics of climate data. Clustering is defined to group similar data to find useful information. But, the existing clustering technique is not efficient for increasing accuracy as well as decreasing prediction time. Depend on motivation, a novel DSFS-MSBBMPDC technique is introduced with two different processes.

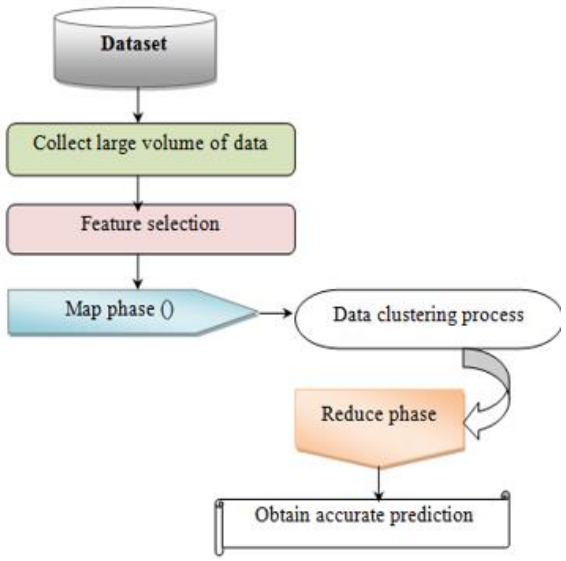

Figure 1 architecture of proposed DSFS-MSBBMPDC technique

Fig 1 presents DSFS-MSBBMPDC for performing the big data prediction by performing the two different processes namely feature selection and clustering. Initially, big Dataset '*D*', features set $\{k_1, k_2, k_3, \ldots . k_n\}$ and number of input data $d_1, d_2, d_3, \ldots . d_m$are composed. DSFS-MSBBMPDC first performs feature selection process using a Decision stump. With the extracted features, clustering procedure is achieved for achieving high prediction accuracy as well as minimum error. DSFS-MSBBMPDC is explained in subsequent subsection.

3.1 Otsuka-Ochiai decision stump based feature selection

First, the Decision Stump Feature Selection is performed in proposed DSFS-MSBBMPDC for choosing more important features to minimize time. Decision stump comprises one-level decision tree. By one internal node, decision tree is linked to terminal node.

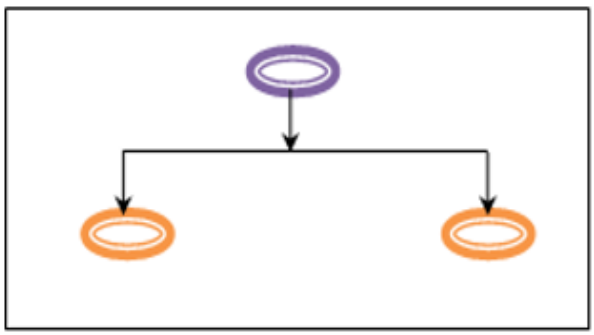

Figure 2 Otsuka-Ochiai decision stump

Figure 2 illustrates the Otsuka-Ochiai decision stump to find the significant features by measuring the similarity analysis. Let us consider the number of features $\{k_1, k_2, k_3, \ldots . k_n\}$. An Otsuka-Ochiai decision stump comprises one-level decision tree where the internal node (i.e. root) was directly linked to leaf node. In the root node, the Otsuka-Ochiai similarity is measured to identify the significant features.

The similarity between the features are estimated as given below,

$$\beta\ (k_i, k_j)\ = \frac{|k_i \cap k_j|}{\sqrt{\sum {k_i}^2}\sqrt{\sum {k_j}^2}} \quad (1)$$

Where'β' indicates an Otsuka-Ochiai similarity coefficient, $\cap$ indicates a mutual dependence between the featuresk_i, k_j,$\sum {k_i}^2$ represents a squared score of k_i, $\sqrt{\sum {k_j}^2}$ indicates a squared score of k_j . Otsuka-Ochiai coefficient provides the similarity value from 0 to 1. The root node measures the similarity with the threshold value. If the similarity is higher than threshold, then it is known as significant feature of prediction. On the other hand, it is removed from the dataset to minimize the complexity. The algorithm process of Otsuka-Ochiai decision stump based feature selection is explained by,

// **Algorithm 1:** Otsuka-Ochiai decision stump feature selection
Input: Dataset, number of features $\omega_1, \omega_2, \omega_3, \ldots . \omega_m$ **Output:** Select significant features
Begin **1.** Collect the number of features $\{k_1, k_2, k_3, \ldots . k_n\}$ **2.** Construct decision tree **3.** **For each** feature 'k_i' and k_j **4.** Measure the similarity '$\beta\ (k_i, k_j)$' **5.** **if** $(\beta\ (k_i, k_j) > T)$ **then** **6.** the feature is said to be relevant **7.** **else** **8.** the feature is said to be irrelevant **9.** **end if** **10.** **end for** **11.** **Select** relevant features **12.** **Remove ir**relevant features **End**

Algorithm 1 illustrates Otsuka-Ochiai decision stump based feature selection is carried out with different processes. Firstly, amount of features is composed via dataset. After that, similarities between the two features are measured. If the similarity is higher than the threshold, then the feature was relevant. On the

other hand, feature was irrelevant. The relevant features are selected for data clustering. This helps to minimize time consumption.

2.2 Mean Shift steepest descent Brown Boost Map Reduce Data clustering process

After the feature selection, the proposed DSFS-MSBBMPDC technique performs the clustering process for improving predictive analytics. The proposed DSFS-MSBBMPDC technique uses the Mean Shift steepest descent Brown Boost Map Reduce Data clustering process for predictive analytics. Shift Brown Boost Map Reduce Data clustering is the ensemble technique used to improve the prediction accuracy by constructing the different weak learners. The weak learner is a base clustering technique that provides less accurate prediction results. In contrast, the boosting ensemble technique provides accurate results. A Map Reduce clustering technique usually partitions the input data and then processes each of them separately. Then output is arranged and input to reduce tasks.

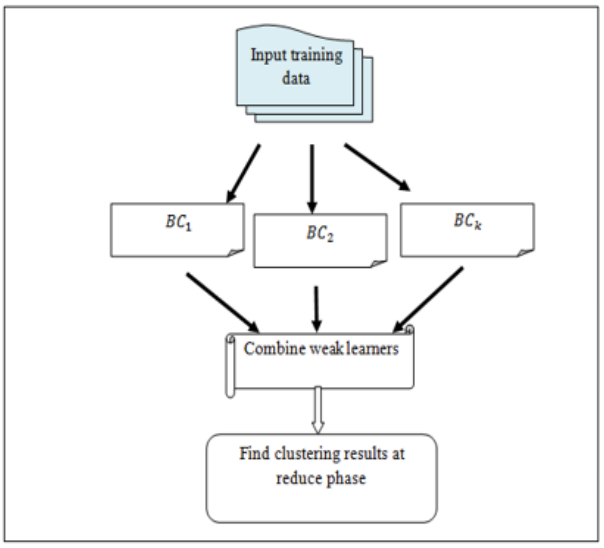

Figure3 flow process of the Mean shift Brown Boost Map Reduce Data clustering

Predicting continental US drought levels using meteorological & soil Figure 3 given above illustrates the flow process of the ensemble clustering technique to efficiently perform predictive analytics. Let us consider the number of training data$d_i = d_1, d_2, \dots, d_m$. Then the proposed ensemble clustering technique comprises a set of 'k' weak clusters $W_1, W_2, W_3, \dots . W_k$. A mean shift clustering technique is used as a weak learner for grouping the input data based on the mean of the clusters. Initially, k number of clusters $W_1, W_2, W_3, \dots . W_k$ and mean value $\beta_1, \beta_2, \beta_3, \dots \beta_k$are initialized in a random manner. For each mean value and the data, the radial basis function kernel is applied for measuring the similarity. The similarity value estimated as given below,

$$\varphi\,[d, \beta_k] = \exp\left[-0.5 * \frac{\|d_i - \beta_k\|^2}{\delta}\right] \quad (2)$$

Where, $\varphi\,[d, \beta_k]$ kernel function, d_i denotes a data, β_k indicates the mean value, δ denotes a deviation from the mean value, and $\|d_i - \beta_k\|^2$ indicates squared Euclidean distance among data as well as mean value of cluster. Depend on distance measure data were correctly grouped in respective clusters.

With the aim of increasing accuracy of clustering, weak learner was summed within strong. Weak learner output were associated and it is expressed as given below,

$$Y = \sum_{i=1}^{n} R(d_i) \quad (3)$$

Where, 'Y' specifies an ensemble clustering results, $R(d_i)$ represents weak learner output. After that, weight is adjusted to all results.

$$Y = \sum_{i=1}^{n} R(d_i)\omega \quad (4)$$

Where 'ω' denotes a weight. After assigning the weight, the generalization error is measured. The error rate is measured as squared differences between the observed and actual results.

$$\boldsymbol{E_{MSE}} = [ac_t - ob_r]^2 \quad (5)$$

Where, E_{MSE} denotes a generalization error, ac_tshows actual results, ob_r indicates observed results. Depend on estimated error value, initial weights get updated. If weak learner is properly grouped, then weight gets reduced. On the other hand, weight is improved. Finally, weak learner results by a lesser error are selected. In order to find lesser error, steepest descent function was utilized.

$$f\,(x) = \arg\min E_{MSE}\,(R(d_i)\,) \quad (6)$$

From (6),$f\,(x)$ denotes the steepest descent function, arg min indicates the argument, E_{MSE} denotes a generalization error, and $R(d_i)$ denotes weak learner results. As a result, accurate clustering was attained by reducing phase. Every data are properly grouped with minimum error. Based on the error rate, accurate prediction is achieved by lesser error.

The step by step procedure of Mean Shift steepest descent Brown Boost Map Reduce Data clustering is explained by,

Algorithm 2: Mean Shift steepest descent Brown Boost Map Reduce Data clustering

Input: Number of data $d_1, d_2, \ldots, d_m$.
Output: Increase prediction accuracy

Begin
1. **For** each d_i
2. Construct 'k' weak clusters 'W_k'
3. Initialize 'k' number of clusters $W_1, W_2, W_3, \ldots . W_k$
4. Initialize mean value $\beta_1, \beta_2, \beta_3, \ldots \beta_k$
5. **For each** 'W_k'
6. **For each** mean 'β_k'
7. Measure the kernel similarity '$\varphi\ [d, \beta_k]$'
8. Cluster the data
9. **End for**
10. **End for**
11. **End for**
12. Combine all weak clusters $Y = \sum_{i=1}^{n} R(d_i)$
13. **For** each weak learner '$R(d_i)$'
14. Assign the weight 'ω'
15. Compute the error '$\boldsymbol{E_{MSE}}$'
16. Update the weight '$\nabla\omega$'
17. Apply steepest descent function '$f\ (x)$'
18. Find the weak learner with minimum error ' $\arg\min E_{MSE}\ (R(d_i))$'
19. **end for**
20. Return (accurate clustering results)

end

Algorithm 2 describes Mean Shift steepest descent Brown Boost Map Reduce Data clustering for accurate prediction. Ensemble method initially constructs 'k' number of clusters. For each cluster, the mean value is initialized. In order to calculate similarity among data and mean value, kernel function is utilized. Depend on similarity measure data were properly grouped within cluster. Then ensemble technique associates weak learner results and measures the generalization error. The steepest descent function is applied to detecting weak learner results by minimum error. Lastly, clustering results with a minimum error are selected as final results.

3. Experimental Scenario

In this section, simulation of DSFS-MSBBMPDC and MR-HDBSCAN* [1], IWKM [2] were developed in Java language by Predict Droughts using Weather & Soil Data. The dataset is collectedfromhttps://www.kaggle.com/cdminix/us-drought-meteorological data/version/5?select=soil_data.csv. It contains the weather as well as soil testing report and it has 4 different files as testing time-series data, training time-series data, validation data, and soil data. The simulation features via soil data and testing time-series data, training time-series data, validation data are combined to obtain the complete set of the dataset. The dataset consists of lakes of data and the select the significant features for the prediction. Among the many data, the amount of data consumed via dataset ranges as 10000-100000. In order to predict the class of drought severity, the target vector feature column is 'score', ranging from 0.0 to 5.0 which are denoted by floats. Based on the score value, different level of Droughts prediction is performed through the ensemble clustering technique.

Table 1 Drought Prediction class based on weather and soil data

Classes	Score value	Drought Severity
1	0.0	No Drought
2	1.0	Abnormally Dry
3	2.0	Moderate Drought
4	3.0	Severe Drought
5	4.0	Extreme Drought
6	5.0	Exceptional Drought

4. Results and discussion

The simulation result of proposed DSFS-MSBBMPDC technique and existing MR-HDBSCAN* [1], IWKM [2] are estimated by prediction accuracy, prediction time, false-positive rate, as well as space complexity versus a number of data.

4.1 Impact of Prediction accuracy

Prediction accuracy is referred as proportion of amount of data accurately grouped in particular cluster to entire amount of input sample data. Therefore, prediction accuracy is measured by,

$$Pre_a = \left[\frac{ACd_i}{d_n}\right] * 100 \qquad (7)$$

From (7), 'Pre_a' denotes prediction accuracy, 'ACd_i' specifies amount of data and 'd_n' indicates entire amount of data. The prediction accuracy was measured in percentage (%).

Table 2 Comparison of Predication accuracy

Number of data	Prediction accuracy (%)		
	DSFS-MSBBMPDC	MR-HDBSCAN*	IWKM
10000	92	89	87
20000	91.5	87.5	85
30000	92	87.66	85.33
40000	93	89.25	87.63
50000	94.4	90.2	86.2
60000	93.66	88.5	86.83
70000	92.85	89.28	87.88
80000	93.87	87.63	85.62
90000	94.44	90	87.55
100000	93	89	87

Pre Table 2 given above provides the comparative analysis of prediction accuracy against amount of data composed via dataset. To conduct fair experiment, the amount of data ranges as 10000-100000. Prediction accuracy of three methods namely proposed DSFS-MSBBMPDC and two other existing methods MR-HDBSCAN* [1], IWKM [2] are observed. DSFS-MSBBMPDC technique of prediction accuracy provides greater performance than the other two existing clustering methods. When considering the 10000 weather and soil data to calculate the prediction accuracy in the first iteration. By applying the DSFS-MSBBMPDC technique, 9200 data are properly grouped and prediction accuracy is 92% whereas 8900 and 8700 data are correctly clustered using MR-HDBSCAN* [1], IWKM [2], and the prediction accuracy are 89% and 87% respectively. For each method, ten dissimilar results are attained. After that, DSFS-MSBBMPDC is compared with existing methods. Finally, average value is taken as the final result. The results indicate that DSFS-MSBBMPDC of prediction accuracy is considerably improved as 5% and 7% compared with MR-HDBSCAN* [1], IWKM [2] respectively.

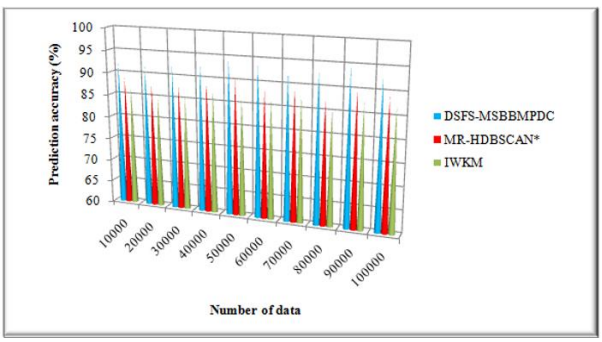

Figure 4 graphical illustration of the prediction accuracy

Prediction accuracy of three methods namely the DSFS-MSBBMPDC technique, MR-HDBSCAN* [1] and IWKM [2] are illustrated in figure 4. From graph, amount of data is consumed in the horizontal direction whereas prediction accuracy was observed in vertical direction. DSFS-MSBBMPDC provides superior performance of existing methods. Significant reason is to apply the Mean Shift steepest descent Brown Boost Map Reduce Data clustering. Ensemble clustering employs mean shift clustering technique for calculating similarity among data as well as mean value. Depend on similarity measure data are properly grouped towards cluster. As a result, different severity levels of the droughts are correctly predicted with higher accuracy.

4.2 Impact of false-positive rate

It is calculated as number of data inaccurately grouped into entire amount of data taken as input. It is measured by,

$$FPR = \left[\frac{NICd_i}{d_n}\right] * 100 \qquad (8)$$

From (8), 'FPR' specifies a false positive rate, '$NICd_i$' indicates number of data inaccurately grouped 'd_n' indicates entire amount of data. It is calculated in percentage (%).

Table 3 Comparison of Prediction accuracy

Number of data	False positive rate (%)		
	DSFS-MSBBMPDC	MR-HDBSCAN*	IWKM
10000	8	11	13
20000	8.5	12.5	15
30000	8	12.33	14.66
40000	7	10.75	12.36
50000	5.6	9.8	13.8
60000	6.33	11.5	13.16
70000	7.14	10.71	12.11
80000	6.12	12.36	14.37
90000	5.55	10	12.44
100000	7	11	13

Table 3 reports the false-positive rate versus DSFS-MSBBMPDC technique, MR-HDBSCAN* [1], and IWKM [2] with the different range of weather and soil data taken from 10000 to 100000. As illustrated in table 3, the experimental assessment is conducted for verifying the accuracy of the proposed DSFS-MSBBMPDC technique and it is compared with the existing clustering technique MR-

HDBSCAN* [1], IWKM [2]. In first iteration, DSFS-MSBBMPDC of false positive rate is 8% and MR-HDBSCAN* [1], IWKM [2] of false-positive rate are 11% and 13%. In all iterations, the rates of proposed and existing methods of false-positive rate were observed by different amount of data. Overall comparison indicates that DSFS-MSBBMPDC of false positive rate is considerably reduced as 38% of MR-HDBSCAN* [1], and 48% of IWKM [2].

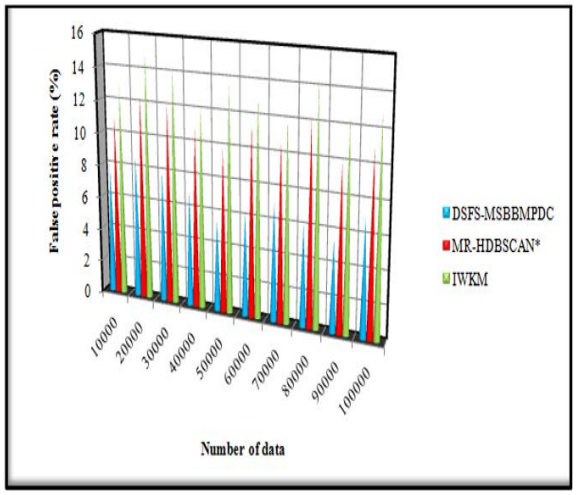

Figure 5 graphical illustration of the false positive rate

Fig 5 illustrates false-positive rate with amount of weather and soil data. Graphical plot illustrates that the performances of the false positive rate with proposed and existing methods. The graphical results clearly reveal that DSFS-MSBBMPDC of false positive rate is considerably reduced by MR-HDBSCAN* [1], IWKM [2]. Steepest descent function in the ensemble clustering technique to find weak learner results by lesser generalization error. This aids for minimizing incorrect clustering process.

4.3 Impact of Prediction time

It is calculated as number of time consumed for prediction through clustering technique. It is formulated by given below,

$$T_p = d_n \times Td_i \qquad (9)$$

Where, 'T_p ' indicates the prediction time, d_n indicates amount of data and 'Td_i' indicates time taken for predicting single data. It is calculated in milliseconds (ms).

Table 4 Comparison of Prediction time

Number of data	prediction time (ms)		
	DSFS-MSBBMPDC	MR-HDBSCAN*	IWKM
10000	42	45	48
20000	48	52	54
30000	51	54	60
40000	60	62	64
50000	65	70	75
60000	72	75	78
70000	77	81	84
80000	80	84	88
90000	83	86	90
100000	86	89	92

Table 4 provides prediction time by amount of data accompanied via dataset. Tabulated results indicate DSFS-MSBBMPDC of prediction time is comparatively minimum than the existing methods. With the consideration of 10000 data, the time taken to predict the droughts severity level was found to be '$42ms$'. However, the prediction of existing MR-HDBSCAN* [1] and IWKM [2] was found to be $45ms$' and$48ms$. The statistical results indicate that the DSFS-MSBBMPDC technique decreases the prediction time. After observed the ten results, the overall time consumption of the DSFS-MSBBMPDC technique is compared with existing methods. DSFS-MSBBMPDC decreases time consumption by 5% and 10% as compared to the MR-HDBSCAN* [1] and IWKM [2]

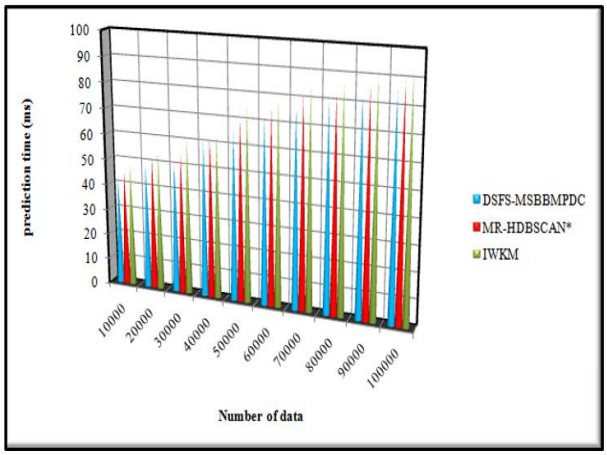

Figure 6 graphical illustration of Prediction time

Figure 6 portrays prediction time using three different clustering methods. From graph, prediction time is directly proportional to

amount data accompanied via dataset. While increasing amount of data, the time consumption for the prediction gets increased for all three methods. Prediction time was relatively lesser by proposed DSFS-MSBBMPDC. Reason behind the minimum time consumption is due to the application of the Otsuka-Ochiai decision stump feature selection. After that, similarities between the two features are measured. If the similarity is higher than the threshold, then the feature is relevant. With selected features, data clustering is performed with minimum time consumption.

4.4 Impact of Space complexity

It is calculated as number of memory space consumed by algorithm for performing big data prediction. Space complexity is measured as given below,

$$S_{com} = d_n \times MSd_i \qquad (10)$$

From (10), 'S_{com}' denotes the space complexity, 'd_n' represents number of data, 'MSd_i' is memory necessary to store single data. Space complexity was measured in terms of Megabytes (MB).

Table 5 Comparison of Space complexity

Number of data	Space complexity (MB)		
	DSFS-MSBBMPDC	MR-HDBSCAN*	IWKM
10000	40	45	49
20000	46	52	56
30000	51	54	60
40000	56	60	64
50000	60	65	70
60000	66	72	78
70000	70	77	81
80000	74	79	84
90000	77	80	86
100000	80	83	88

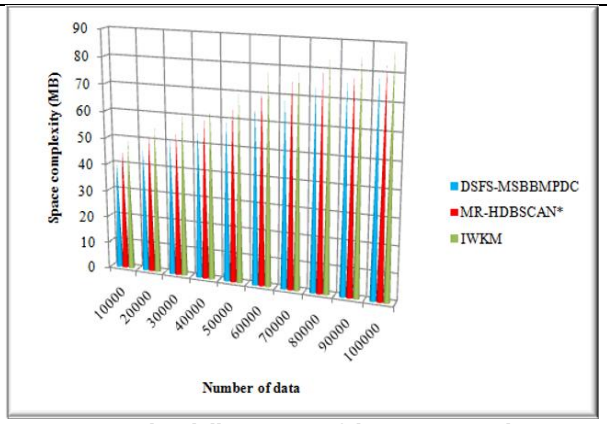

Figure 7 graphical illustration of the space complexity

Table 5 and figure 7 illustrate performance results of space complexity by amount of data. From table and graph, DSFS-MSBBMPDC of space complexity is enhanced as improving amount of data. But comparatively, overall memory space of the DSFS-MSBBMPDC technique is found to be decreased than the existing methods. With the consideration of 10000 data, the space consumption for prediction is found to be '$40MB$'. However, the space consumption for prediction using MR-HDBSCAN* [1] and IWKM [2] was found to be '$45MB$' and $49MB$. DSFS-MSBBMPDC reduces space complexity. After getting the ten results, the average value is taken into the consideration. The average value indicates DSFS-MSBBMPDC of space complexity is reduced as 7% compared with MR-HDBSCAN* [1], 14% when compared to IWKM [2] respectively. This is due to choose significant features via dataset as well as removes the other features.

5. Conclusion

A novel ensemble clustering algorithm called the DSFS-MSBBMPDC is developed for improving prediction accuracy with big data. Partitioning of big data within dissimilar clusters support for decreasing time as well as improve accuracy in big data applications. The DSFS-MSBBMPDC technique performs the Otsuka-Ochiai decision stump feature selection to select the significant relevant features. After that, the mean shift steepest descent brown boost map reduce data clustering process is carried out to partition the data into distinct clusters. The prediction accuracy was enhanced as well as prediction time was reduced. The experiments are performed by amount of data

and compare DSFS-MSBBMPDC with two conventional clustering algorithms. The observed numerical results have confirmed DSFS-MSBBMPDC has enhanced performance in prediction accuracy, false plosive rate, prediction time, and space complexity. Besides, the numerical analysis of accuracy is higher and lesser prediction time than the conventional techniques.

References

Joelson Antônio dos Santos, Talat Iqbal Syed, Murilo C. Naldi, Ricardo J. G. B. Campello, Joerg Sander, "Hierarchical Density-Based Clustering Using MapReduce", IEEE Transactions on Big Data, Volume 7, Issue 1, 2021, Pages 102 – 114

Qian Taoa,Chunqin Gua, Zhenyu Wanga, Daoning Jiang, "An intelligent clustering algorithm for high-dimensional multiview data in big data applications", Neurocomputing, Elsevier, Volume 393, 2020, Pages 234-244

Zahra Karevan and Johan A.K. Suykens, "Transductive LSTM for time-series prediction: An application to weather forecasting", Neural Networks, Elsevier, Volume 125, 2020, Pages 1-9

Weijia Lu, "Improved K-Means Clustering Algorithm for Big Data Mining under Hadoop Parallel Framework", Journal of Grid Computing, Springer, Volume 18, 2020, Pages 239-250

Ashish Kumar Tripathi, Kapil Sharma & Manju Bala, "Dynamic frequency based parallel k-bat algorithm for massive data clustering (DFBPKBA)", International Journal of System Assurance Engineering and Management, Springer, Volume 9, 2018, Pages 866-874

Rasim M. Alguliyev, Ramiz M. Aliguliyev, Lyudmila V. Sukhostat, "Parallel batch k-means for Big data clustering", Computers & Industrial Engineering, Elsevier, Volume 152, 2021, Pages 1-11

Rasim M. Alguliyev, Ramiz M. Aliguliyev, Lyudmila V. Sukhostat, "Weighted consensus clustering and its application to big data", Expert Systems with Applications, Elsevier, Volume 150, 2020, Pages 1-15

G. HimaBindu, Ch. Raghu Kumar, Ch. Hemanand, N. Rama Krishna, "Hybrid clustering algorithm to process big data using firefly optimization mechanism", Materials Today: Proceedings, Elsevier, 2020, Pages 1-5

Michele Ianni, Elio Masciari, Giuseppe M. Mazzeo, Mario Mezzanzanica,Carlo Zaniolo, "Fast and effective Big Data exploration by clustering", Future Generation Computer Systems, Elsevier, Volume 102, 2020, Pages 84-94

Sunil Kumar and Maninder Singh, "A novel clustering technique for efficient clustering of big data in Hadoop Ecosystem", Big Data Mining and Analytics, Volume 2, Issue 4, 2019, Pages 240 – 247

Zhonghao Xue and Hongzhi Wang, "Effective density-based clustering algorithms for incomplete data", Big Data Mining and Analytics, Volume 4, Issue3, 2021, Pages 183 - 194

Donghui Yan, Yingjie Wang, Jin Wang, Guodong Wu, and Honggang Wang, "Fast Communication-efficient Spectral Clustering Over Distributed Data", IEEE Transactions on Big Data, Volume 7, Issue 1, 2021, Pages 158 – 168

Hongyan Ma, "The role of clustering algorithm-based big data processing in information economy development", PLoS ONE, Volume 1, Issue 3, 2021, Pages 1-14

Yinghua Shen, Witold Pedrycz,Yuan Chen, Xianmin Wang, Adam Gacek, "Hyperplane Division in Fuzzy C-Means: Clustering Big Data", IEEE Transactions on Fuzzy Systems, Volume 28, Issue 11, 2020, Pages 3032 - 3046

Jungrim Kim, Mincheol Shin, Jeongwoo Kim, Chihyun Park, Sujin Lee, Jaemin Woo1Hyerim Kim, Dongmin Seo, Seokjong Yu, Sanghyun Park, "CASS: A distributed network clusteringalgorithm based on structure similarity forlarge-scale network", . PLoS ONE, Volume 13, Issue 10, 2018, Pages 1-22

Safanaz Heidari, Mahmood Alborzi, Reza Radfar, Mohammad Ali Afsharkazemi & Ali Rajabzadeh Ghatari, "Big data clustering with varied density based on MapReduce", Journal of Big Data, Springer, Volume 6, 2019, Pages 1-16

Phongphun Kijsanayothin, Gantaphon Chalumporn and Rattikorn Hewett, "On using MapReduce to scale algorithmsfor Big Data analytics: a case study", Journal of Big Data, Springer, Volume 6, 2019, Pages 1-20

Ye Shi, Chin-Teng Lin, Yu-Cheng Chang, Weiping Ding, Yuhui Shi, Xin Yao, "Consensus Learning for Distributed Fuzzy Neural Network in Big Data Environment", IEEE Transactions on Emerging Topics in Computational Intelligence, Volume 5, Issue 1, 2021, Pages 29 – 41

Xiangping Xu, Jun Li, MengChu Zhou, Jun Xu, Jinde Cao, "Accelerated Two-Stage Particle Swarm Optimization for Clustering Not-Well-Separated Data", IEEE Transactions on Systems, Man, and Cybernetics: Systems, Volume 50, Issue 11, 2020, Pages 4212 – 4223

Marc Hüsch, Bruno U. Schyska, Lueder von Bremen, "CorClustST—Correlation-based clustering of big spatio-temporal datasets", Future Generation Computer Systems, Elsevier, Volume 110, 2020, Pages 610-619