Industrial Engineering Journal
ISSN: 0970-2555
Volume : 52, Issue 5, No. 3, May : 2023

MAPREDUCE EXPECTATION–MAXIMIZATION CLUSTERING WITH ENSEMBLED BOOTSTRAP AGGREGATION FOR BIG DATA ANALYTICS

K.M.PADMAPRIYA Assistant professor in Computer Science, SSM College of Arts & Science, Tamilnadu ,India.
Dr. B.ANANDHI Associate Professor & Head, Department of Computer Science, Vellalar College for Women, Tamilnadu. India.
Dr. S.BALAMOHAN Principal,SSM College of Engineering, Komarapalyam, Namakkal Dt.Tamilnadu,
*[1]padmapriya.kmp@gmail.com[2]ananthibalamohan@gmail.com [3]balamohussm@gmail.com

ABSTRACT

Clustering is an important problem to be resolved for data analysis and knowledge discovery. In the context of big data, grouping of similar data is a difficult issue due to the huge quantity of data. A lot of clustering algorithms have been designed in conventional works. However, the computational cost of existing clustering was higher whenever the size of data gets larger. In addition to that, the accuracy of existing big data clustering is not adequate. In order to overcome the above drawbacks, Bootstrap Aggregated MapReduce Expectation–Maximization Clustering (BAMEC) Technique is proposed. Initially, BAMEC technique takes the big Brazilian E-Commerce Public Dataset as input. After getting input, a BAMEC technique generates the number of bootstrap samples from an input big dataset. Subsequently, a BAMEC technique designs the 'N' number of MapReduce based Expectation–Maximization (MEM) clustering for each constructed bootstrap samples. Next, a BAMEC technique combines the outputs of all MEM clustering's and consequently applies voting method. At last, BAMEC technique exactly clusters the similar data by considering the majority vote's results. Thus, BAMEC technique enhances the efficiency of clustering with a lower amount of time for analyzing big data. The BAMEC technique performs the experimental process using metrics such as clustering accuracy, computational cost and false alarm rate and space complexity. The experimental result illustrates that the BAMEC technique is able to enhance the clustering accuracy and also reduces the computational cost of big data analytics as compared to state-of-the-art works.

Keywords: Bagging, Big Data, Cluster Centroid, Expected Probability, Hadoop MapReduce, Majority Vote's, Mapping, Reducing

1. INTRODUCTION

Big data the term represents datasets with high volume, variety, velocity and value. The big datasets present problems with storage, analysis, and visualization. Huge amounts of data are collected from diverse sources and therefore there is a high demand for methods to efficiently analyze big data. Many clustering algorithms are developed for processing big data with the help of different data mining techniques. However, existing clustering algorithms are not suitable when considering a large size of data as input. Besides, the computational cost of conventional clustering algorithms was higher as it requires more time to build the groupings. In order to address the above drawbacks, the Bootstrap Aggregated MapReduce Expectation–Maximization Clustering (BAMEC) Technique is introduced in this research work.

A Weighted Consensus Fuzzy Clustering (WCFC) was presented in [1] to reduce the time of clustering for handling big data with application of RHadoop's parallel processing MapReduce framework. But, the number of data wrongly clustered using WCFC was more. Spectral Ensemble Clustering (SEC) was presented in [2] to decreases the algorithmic complexity of big data clustering. However, the computational complexity needed to cluster big volume of data was not minimized.

Adaptive Dual-Similarity Clustering Ensemble Algorithm was presented in [3] with the aim of improving the clustering performance of data. However, the false positive rate of data clustering was poor. MapReduce-based k-prototypes clustering were introduced in [4] in order to enhance the clustering efficiency of large scale data. But, the time complexity involved during mixed large scale data clustering was not solved.

A different Ensemble technique designed for big data analytics was analyzed in [5]. Two clustering validity indices were presented in [6] to group a huge capacity of data with minimal time. However, clustering accuracy was poor. An enhanced k-prototypes clustering algorithm was employed in [7] for grouping assorted numeric and categorical data by determining the similarity. But, the scalability of this method was not addressed when taking big dataset as input.

The MapReduce-based fuzzy c-means clustering algorithm was introduced in [8] to solve the issue of parallelization during the clustering process of large volume of data. However clustering accuracy was not higher. A novel density-based clustering method was developed in [9] to perform clusters with varying densities and parallelizing the algorithm with MapReduce. But, space complexity of this algorithm was higher. Parallel Spectral Clustering was carried out in [10] to resolve the memory and computation problems on distributed computers. However, the clustering performance was poor.

In order to resolve the above mentioned conventional problems in big data analytics, BAMEC technique is developed in this research work. The key contributions of BAMEC technique are shown in below,

- To reduce the amount of computational cost required for efficient big data analytics when compared to state-of-the-art works, MapReduce based Expectation–Maximization (MEM) clustering algorithm is proposed in BAMEC technique to clusters the big data. The MEM clustering groups similar kind of data in large dataset via utilizing the parallelism among clusters. This process results in minimized the computational cost.

- To achieve enhanced clustering accuracy for big data, Bagging is employed in MEM clustering algorithm. The Bootstrap Aggregation (Bagging) is an ensemble method where it combines the predictions from numerous MEM clustering together to construct strong cluster.

The rest of the paper is planned as follows. Section 2 shows the related works. In Section 3, the proposed BAMEC technique is explained with the help of architecture diagram. In Section 4, Simulation settings are described and the experimental result of BAMEC technique is presented in Section 5. Section 6 depicts the conclusion of the paper.

2. RELATED WORKS

A scalable and flexible clustering scheme was designed in [11] depends on boundary information to get better clustering accuracy and time complexity. MapReduce programming was designed in [12] to perform big data assessment with higher accuracy. Parallel power iteration clustering was introduced in [13] in order to lessen the computation and communication costs of big data grouping.

Hybrid Feature Clustering (HFC) method was employed in [14] to increase the accuracy and to diminish the complexity of high-dimensional data. A single-pass solution depends on MapReduce was designed in [15] with the help of reclustering technique to obtain higher quality of clustering results and minimizing execution time.

A novel consensus clustering algorithm was presented in [16] with the application of Minkowski distance to lessen the false positive rate. A collaborative multi-view clustering based on K-means hypothesis was introduced in [17] to get better performance for clustering of data. A Parallel swarm intelligence strategies were developed in [18] for accomplishing large-scale clustering with MapReduce function.

The similarities and differentiation between the *K*-means algorithm and the Canopy algorithm's MapReduce execution were presented in [19] for clustering analysis of big data sets. A novel criterion was used in [20] to minimize the processing time and to improve the quality of *k*-means cluster algorithms in big data realms.

3. BAGGING ENSEMBLED MAPREDUCE EXPECTATION–MAXIMIZATION CLUSTERING TECHNIQUE

The BAMEC Technique is designed by integrating the Hadoop MapReduce function in Expectation–Maximization Clustering and Bootstrap aggregation method to improve accuracy of clustering huge size of data. The BAMEC Technique designs MapReduce based Expectation–Maximization (MEM) clustering algorithm on the contrary to state-of-the-art works to clusters the big volume of data in an input dataset with minimal time. In BAMEC Technique, MEM clustering algorithm is proposed by combining the Hadoop MapReduce function in Expectation–Maximization Clustering. The MEM clustering algorithm introduced for processing big data by exploiting the parallelism among a cluster and thereby minimizing the computational cost involved during big data clustering process. On the contrary to conventional works, Hadoop MapReduce function contains three key features such as simple programming framework, linear scalability and fault tolerance. These features make MEM clustering efficient for big data processing with lower time complexity. The MEM clustering algorithm includes two main phases namely mapping and reducing.

During the mapping phase, the expected likelihood probability is determined between the cluster centroids and data. Followed by, maximum a posteriori function is employed in reducing phase for maximizing the expected probability to discover the exact cluster members. This helps for MEM clustering algorithm to group similar data with a minimal amount of time. To further enhance the clustering accuracy of big data, the bagging method is applied in BAMEC Technique. The Bagging refers 'bootstrap aggregation 'which is a meta-algorithm where it takes 'X' number of bootstrap samples from the input dataset and trains the MEM clustering on those samples. The final result is obtained based on majority votes of base MEM clustering outputs. The architecture diagram of BAMEC techniques is demonstrated in below Figure 1.

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 52, Issue 5, No. 3, May : 2023

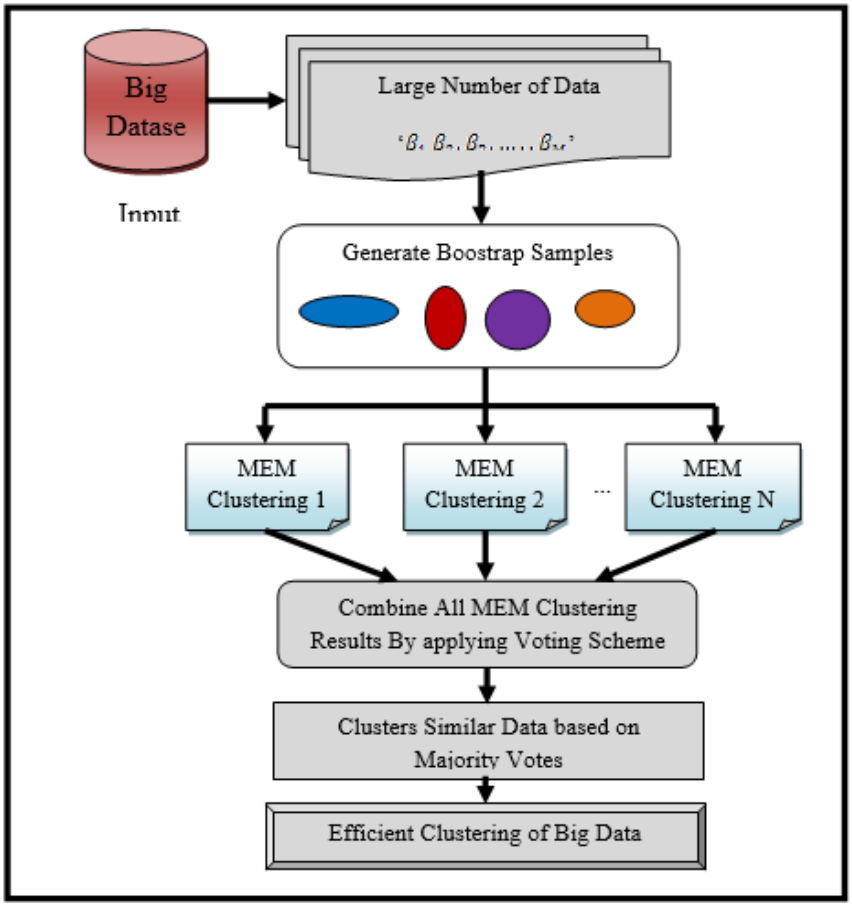

Figure 1 Architecture Diagram of BAMEC Techniques for Big Data Clustering and Analytics

Figure 1 presents the overall processes of BAMEC techniques to attain higher big data analytics performance. As demonstrated in the above architecture diagram, a BAMEC technique at first gets the big dataset i.e. Brazilian E-Commerce Public Dataset as input. This big dataset includes of a huge number of data represented as '$\beta_1, \beta_2, \beta_3, \ldots\ldots \beta_M$'. After taking input, a BAMEC technique creates the bootstrap samples from an input dataset. Then, a BAMEC technique constructs the 'N' number of MapReduce based Expectation–Maximization (MEM) clustering for each bootstrap samples. Followed by, a BAMEC technique aggregates the results of all MEM clustering's and applies voting scheme. Finally, BAMEC technique groups the similar data together based on majority vote's results. From that, proposed BAMEC technique effectively performs big data clustering process with a minimal amount of computational cost. The detailed processes of BAMEC technique is described in below.

In BAMEC technique, the MapReduce based Expectation–Maximization (MEM) clustering is proposed with the aim of carried outing the big data clustering process with a lower computational cost. The MEM clustering algorithm is developed with the application of Hadoop MapReduce function in Expectation–Maximization Clustering. The MEM clustering is a parallel programming framework proposed to process large scale data across a clusters. The Hadoop MapReduce function is employed in MEM as it characterized by its high transparency which allows to parallelize cluster process in a simple and easy manner.

In MEM, clustering of big data is parallelized using two phases namely mapping and reducing. Each phase in MEM clustering includes '$< key/value >$' pairs as input and output. Here, key denotes a cluster centroid and value indicates an expected probability of data. The mapping phase takes in parallel each key/value pair and generates one or more intermediate key/value pairs

through measuring the expected probability between data and cluster centroid. The reducing phase generates final clustering result by maximizing an expected probability between data and cluster centroid. Thus, MEM significantly accomplishes big data clustering with a minimal amount of time consumption. The flow process of MEM clustering is depicted in Figure 2.

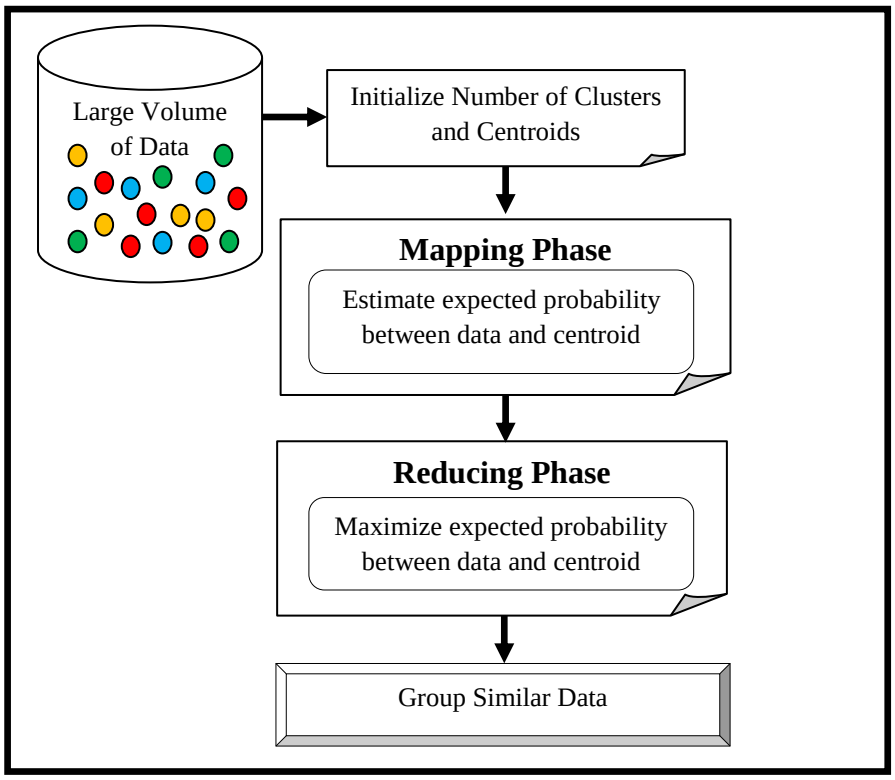

Figure 2 MEM Clustering Process for Big Data

Figure 2 explains the flow process of MEM in order to reduce the computational cost of clustering the big data. As presented in the above diagram, the MEM clustering algorithm at first takes a big volume of data as input. Then, the MEM clustering algorithm initializes the number of clusters and cluster centroids. Subsequently, the expected probability between each centroid and data is estimated in the mapping phase. In the reducing phase, the maximum a posteriori is utilized in order to maximize expected probability between each centroid and data and thereby grouping the similar kind of data together with minimal time.

The algorithmic process of MEM clustering is explained in below,

// MapReduce based Expectation–Maximization Clustering Algorithm
Input: a Larger number of Data in Dataset '$DS = \beta_1, \beta_2, \beta_3, \ldots\ldots \beta_M$'
Output: Grouping similar data with a minimal computational cost
Step 1:Begin
Step 2: Initialize the number of clusters '$\alpha_1, \alpha_2, \alpha_3, \ldots\ldots \alpha_N$' and
Step 3: Define cluster centroids '$\gamma_1, \gamma_2, \gamma_3, \ldots\ldots \gamma_N$'
Step 4: **For** each data '$\beta_i \in DS$'
Step 5: **For** each cluster centroid 'γ_i' **do**
Step 6: Measure'$Exp\ \{P\ (\gamma_i | \beta_i)\}$' using (1)
Step 7: Performing mapping tasks using (2)
Step 8: Determine 'ϑ_{MAP}' using (3)
Step 9: Reducing Phase group the data into a particular cluster
Step 10: **End for**

Step 11: End for
Step 12:End

Algorithm 1 MapReduce based Expectation–Maximization Clustering

Algorithm 1 shows the step by step processes of MEM clustering algorithm to attain the minimal computational cost to perform big data clustering. By using the above algorithmic process, MEM clustering algorithm reduces the amount of time required to cluster the huge size of data. As a result, the MEM clustering algorithm achieves a lower computational cost for grouping the similar kind of data together in an input dataset. The MEM clustering algorithm groups the data into corresponding clusters. But, clustering accuracy of MEM clustering algorithm was not higher because of the probability results. In order to obtain the higher accuracy for clustering large volume of data, bagging method i.e. bootstrap aggregation is employed in BAMEC technique.

In the proposed BAMEC technique, bootstrap aggregation is a machine learning ensemble meta-algorithm. The bootstrap aggregation technique is applied in proposed work to improve the stability and accuracy of big data clustering and also to avoid overfitting. Bootstrap Aggregation is an ensemble method where multiple clusters are trained using random sampling with replacement to produce diverse training sets for each one (i.e. Bootstraps). Each input data is clustered according to the majority votes predicted by the ensemble. In this manner, BAMEC technique avoids the overfitting and smooth out data set variability during the process of big data clustering. Initially, diverse training bootstrap samples sets are constructed from an input dataset to improve clustering results of big data analytics. The generation process of bootstrap samples is demonstrated in below.

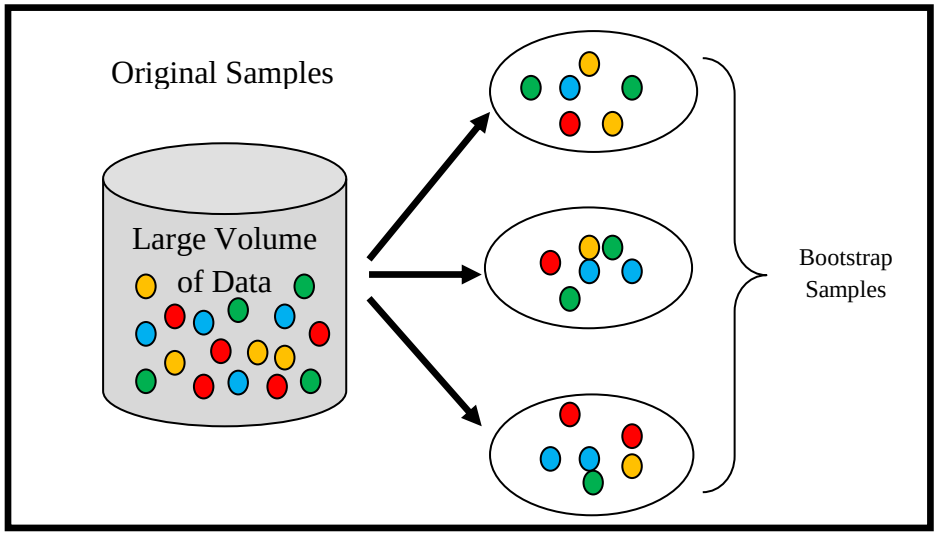

Figure 3 Bootstrap Samples Generation Process

Figure 3 shows the formation of training bootstrap samples to increase the accuracy of MEM clustering algorithm for an input big dataset. Bootstrapping is a procedure of constructing random samples with a replacement for estimating sample statistics. As depicted in the above figure, BAMEC technique chooses 'X' data with replacement from an original sample 'M' to form bootstrap samples. A bootstrap sample contains a few duplicate data, as the sampling is performed with replacement. The generated bootstrap samples supports for BAMEC technique to improve clustering accuracy, decrease variance and bias and also to increases stability. After that, MEM clustering is employed as a base learner for training bootstrap samples and then grouping is performed based on majority votes.

The algorithmic processes of BAMEC technique are described in below.

// Bagging Aggregated Ensembled MapReduce Expectation–Maximization Clustering Algorithm

Input: Larger number of Data in Dataset '$DS = \beta_1, \beta_2, \beta_3, \ldots . \beta_M$'
Output: Enhanced Clustering Accuracy for Big Data
Step 1: Begin
Step 2: For each input big dataset
Step 3: Create Bootstrap Samples
Step 4: Construct 'N' number of MEM clustering results
Step 5: Aggregate All the MEM clustering results using (5)
Step 6: Apply voting scheme using (6)
Step 7: Group similar data according to majority votes using (7)
Step 8: End For
Step 9:End

Algorithm 2 Bagging Aggregated Ensembled MapReduce Expectation–Maximization Clustering

Algorithm 2 explains the step by step processes of BAMEC technique to get enhanced accuracy for carried outing the big data clustering process. By using the above algorithmic steps, BAMEC technique accurately groups the more similar data together into diverse clusters with minimal time. Hence, BAMEC technique attains better performance in terms of clustering accuracy, computational cost, and false alarm rate and space complexity for analyzing the big data as compared to state-of-the-art works.

4. EXPERIMENTAL SETTINGS

In order to determine the performance of the proposed, BAMEC technique is implemented in Java Language using Brazilian E-Commerce Public Dataset [21]. The Brazilian E-Commerce Public Dataset is a retail dataset that comprises of anonymized orders made at Olist (100k orders) from 2016 to 2018 made at numerous marketplaces. The BAMEC technique takes a different number of data in the range of 1000-10000 from Brazilian E-Commerce Public Dataset to perform the experimental process. The efficiency of BAMEC technique is measured in terms of clustering accuracy, computational cost, and false alarm rate and space complexity. The performance results of BAMEC technique are compared against two conventional methods namely Weighted Consensus Fuzzy Clustering (WCFC) [1] and Spectral Ensemble Clustering (SEC) [2].

5. RESULT AND DISCUSSIONS

In this section, the comparative result analysis of BAMEC technique is presented. The experimental result of BAMEC technique is compared with WCFC [1] and SEC [2] respectively using below parameters such as clustering accuracy, computational cost, and false alarm rate and space complexity with the assist of below tables and graphical representation.

5.1 Measurement of Clustering Accuracy

Clustering accuracy 'A' estimated as the ratio of number of data correctly clustered to the total number of data. The clustering accuracy is mathematically evaluated as,

$$A = \frac{m_{cc}}{M} * 100 \quad (1)$$

From equation (1), 'm_{cc}' signifies number of correctly grouped data whereas 'M' point outs a total number of data considered to conduct experimental process. The clustering accuracy is determined in terms of percentage (%). When a clustering accuracy is higher, the technique is said to be more efficient.

The BAMEC technique is implemented in Java language by considering varied number of data in the range of 1000-10000 to calculate the accuracy of big data clustering. When performing experimental evaluation using 7000 data from Brazilian E-Commerce Public Dataset, proposed

BAMEC technique gets 94 % clustering accuracy whereas conventional algorithms WCFC [1] and SEC [2] achieves 82 % and 78 % respectively. Thus, it significant that the clustering accuracy using proposed BAMEC technique is higher than other works. The comparative result analysis of clustering accuracy is shown in below Table 1.

Table 1 Experimental Result of Clustering Accuracy

Number of data (M)	Clustering Accuracy (%)		
	BAMEC	WCFC	SEC
1000	89	86	85
2000	90	88	86
3000	93	85	80
4000	90	86	80
5000	91	88	82
6000	94	85	80
7000	94	82	78
8000	91	86	79
9000	95	86	83
10000	94	83	81

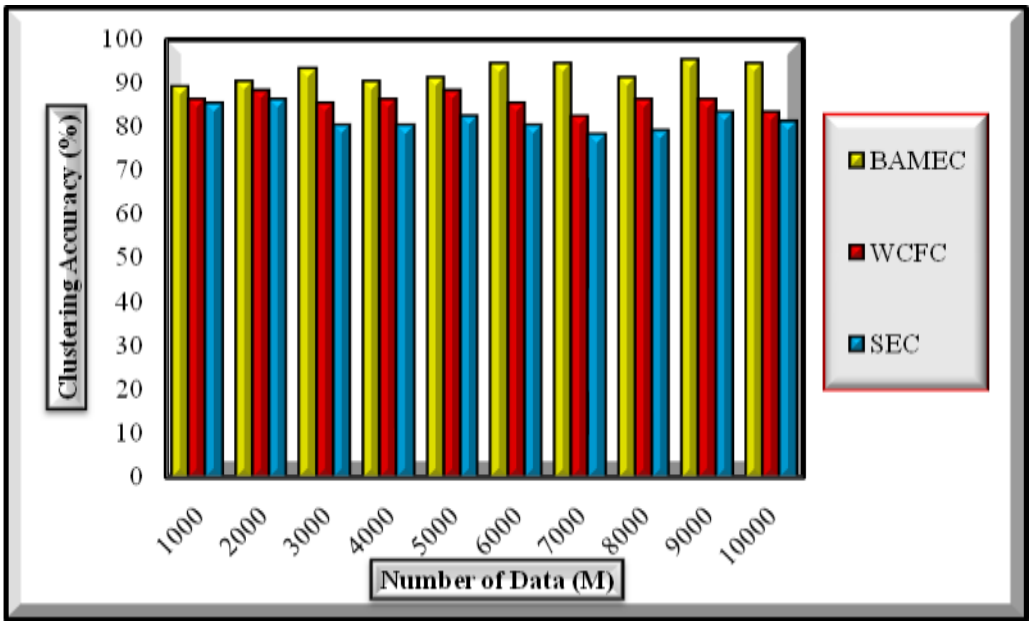

Figure 4 Performance Result of Clustering Accuracy Vs Number of Data

Figure 4 illustrates the experimental result of clustering accuracy versus a diverse number of data using three methods namely BAMEC technique, WCFC [1] and SEC [2]. As presented in the above figure, BAMEC technique achieves higher clustering accuracy and thereby increases the performance of big data analytics when compared to WCFC [1] and SEC [2]. This is owing to the application of MEM clustering and bootstrap aggregation in BAMEC technique on the contrary to conventional methods. With the concepts of bootstrap aggregation, BAMEC technique enhances the performance of MEM clustering to accurately group the data by considering majority vote results. This helps for MGDGAC technique to increases the ratio of a number of data correctly clustered when compared to other existing works. Therefore, MGDGAC technique enhances the clustering accuracy of big data analytics by 8 % when compared to WCFC [1] and 13 % when compared to SEC clustering [2] respectively.

5.2 Measurement of Computational Cost

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 52, Issue 5, No. 3, May : 2023

Computational Cost 'CC' determines the time needed for clustering similar data. The computational cost is mathematically calculated as.

$$CC = M * T_{CS} \quad (2)$$

From equation (2), 'T_{CS}' refers a time utilized to cluster a single data and 'M' denotes a total number of data. The computational cost is measured in terms of milliseconds (ms). When a computational cost is lower, the technique is said to be more effective.

To determine the computational cost of big data clustering, the BAMEC technique is implemented in Java language with the assist of a diverse number of data in the range of 1000-10000. When conducting experimental process using 9000 data from Brazilian E-Commerce Public Dataset, proposed BAMEC technique attains 80 ms computational cost whereas state-of-the-art works WCFC [1] and SEC [2] gets 86 ms and 90 ms respectively. Hence, it considered that the computational cost using proposed BAMEC technique is lower than other existing works. The performance result analysis of computational cost is portrayed in below Table 2.

Table 2 Experimental Result of Computational Cost

Number of data (M)	Computational Cost (ms)		
	BAMEC	WCFC	SEC
1000	20	27	30
2000	24	34	40
3000	27	36	39
4000	32	40	48
5000	50	56	60
6000	63	68	72
7000	69	81	84
8000	78	85	88
9000	80	86	90
10000	88	93	96

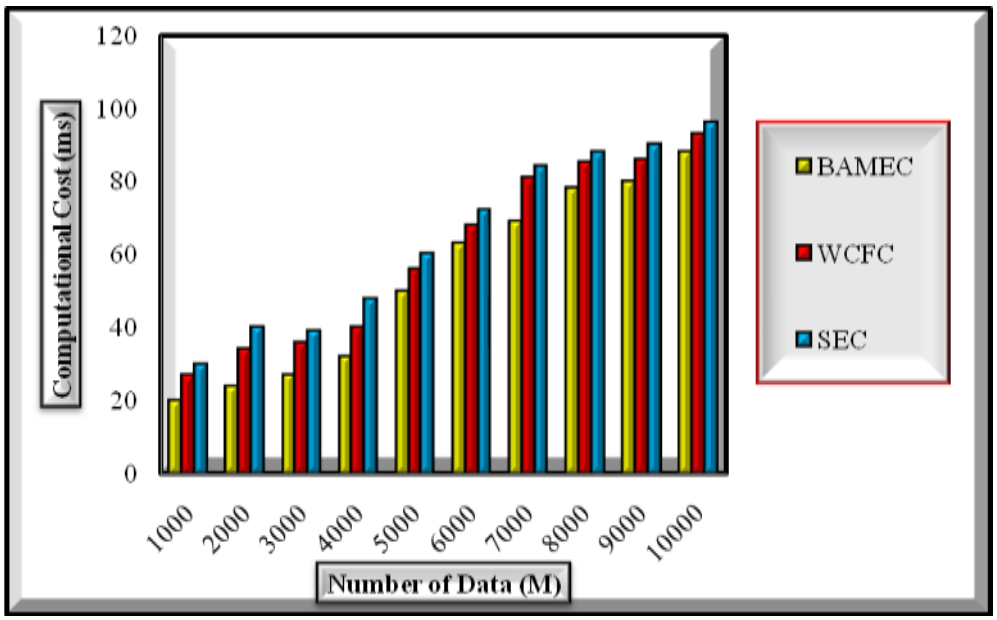

Figure 5 Performance Result of Computational Cost Vs Number of Data

Figure 5 demonstrates the comparative results of computational cost versus a dissimilar number of data using three methods namely BAMEC technique, WCFC [1] and SEC [2]. As demonstrated in the above figure, BAMEC technique attains a lower computational cost for big data

analytics when compared to WCFC [1] and SEC [2] respectively. This is because of the application of MEM clustering in BAMEC technique on the contrary to existing methods. By using the algorithmic processes of MEM clustering, BAMEC technique parallelizes big data clustering process using two phases i.e. mapping and reducing. This assists for BAMEC technique to employs a minimum amount of time for efficiently grouping similar data in a given dataset when compared to other works. Thus, BAMEC technique decreases the computational cost of big data analytics by 13 % when compared to the WCFC framework [1] and 22 % when compared to SEC clustering [2] respectively.

5.3 Measurement of False Alarm Rate

False Alarm Rate 'FAR' computed as the ratio of number of data wrongly clustered to the total number of data. The false alarm rate is mathematically determined using below,

$$FAR = \frac{M_{ic}}{M} * 100 \quad (3)$$

From equation (3), 'M_{ic}' indicates a number of data wrongly clustered and 'M' signifies a total number of data. The false alarm rate is calculated in terms of percentage (%).When a false alarm rate is minimal, the technique is said to be more effectual.

For evaluating the false alarm rate during the clustering process of big data size, the BAMEC technique is implemented in Java language by using different numbers of data in the range of 1000-10000. When carried outing experimental process using 8000 data from Brazilian E-Commerce Public Dataset, proposed BAMEC technique attains 9 % false alarm rate whereas existing WCFC [1] and SEC [2] acquires 14 % and 21 % respectively. As a result, it is expressive that the false alarm rate using proposed BAMEC technique is minimal than other existing works. The experimental result analysis of the false alarm rate is demonstrated in below Table 3.

Table 3 Experimental Result of False Alarm Rate

Number of data (M)	False Alarm Rate (%)		
	BAMEC	WCFC	SEC
1000	11	13	16
2000	10	12	14
3000	7	15	20
4000	10	14	20
5000	9	12	18
6000	6	15	20
7000	6	18	22
8000	9	14	21
9000	5	14	17
10000	6	17	19

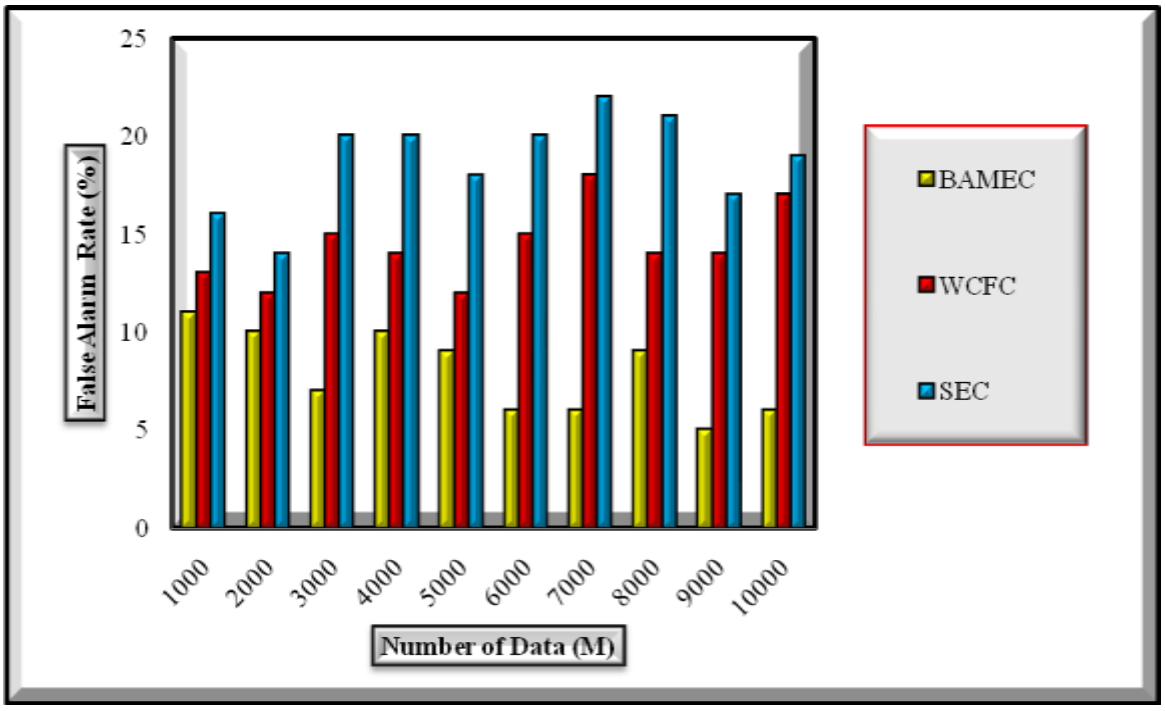

Figure 6 Performance Result of False Alarm Rate Vs Number of Data

Figure 6 depicts the experimental result analysis of false alarm rate versus a varied number of data using three methods namely BAMEC technique, WCFC [1] and SEC [2]. As shown in the above figure, BAMEC technique gets a minimal false alarm rate for big data analysis when compared to WCFC [1] and SEC [2] respectively. This is due to the application of MEM clustering and bootstrap aggregation in BAMEC technique on the contrary to existing algorithms. With the help of bootstrap aggregation concepts, BAMEC technique lessens the inaccurate clustering of big data through combines the 'N' number of MEM clustering outputs. The BAMEC technique builds the strong clustering result according to majority vote which resulting in efficient clustering of huge volume of data. This supports for BAMEC technique to minimize the number of data wrongly clustered when compared to other existing [1] and [2]. Thus, BAMEC technique decreases the false alarm rate of big data analytics by 43 % as compared to WCFC framework [1] and 56 % as compared to SEC clustering [2] respectively.

5.4 Measurement of Space Complexity

Space complexity 'C' measures the amount of memory space needed to store the clustered data. The space complexity is mathematically calculated using below,

$$C = M * m_{SS} \quad (4)$$

From equation (4), 'm_{SS}' denotes a memory required to store single clustered data and 'M' point outs a total number of data. The space complexity is estimated in terms of mega bytes (MB). When a space complexity is lower, the technique is said to be more effective.

For determining the space complexity of large data clustering process, the BAMEC technique is implemented in Java language with the aid of dissimilar numbers of data in the range of 1000-10000. When conducting the experimental work using 5000 data from Brazilian E-Commerce Public Dataset, proposed BAMEC technique obtains 36 MB space complexity whereas WCFC [1] and SEC [2] gets 45 MB and 47 MB respectively. For that reason, it is significant that the space complexity using proposed BAMEC technique is minimal than other works. The tabulation result analysis of space complexity is presented in below Table 4.

Table 4 Experimental Result for Space Complexity

Number of data (M)	Space Complexity (MB)		
	BAMEC	WCFC	SEC
1000	22	28	31
2000	26	33	36
3000	29	35	39
4000	32	39	42
5000	36	45	47
6000	39	48	50
7000	42	52	55
8000	47	55	59
9000	51	57	61
10000	55	63	66

Figure 7 presents the impact of space complexity versus a diverse number of data using three methods namely BAMEC technique, WCFC [1] and SEC [2]. As depicted in the above figure, BAMEC technique attains minimal space complexity as compared to WCFC [1] and SEC [2] respectively. This is due to the application of bootstrap aggregation in BAMEC technique on the contrary to conventional work.

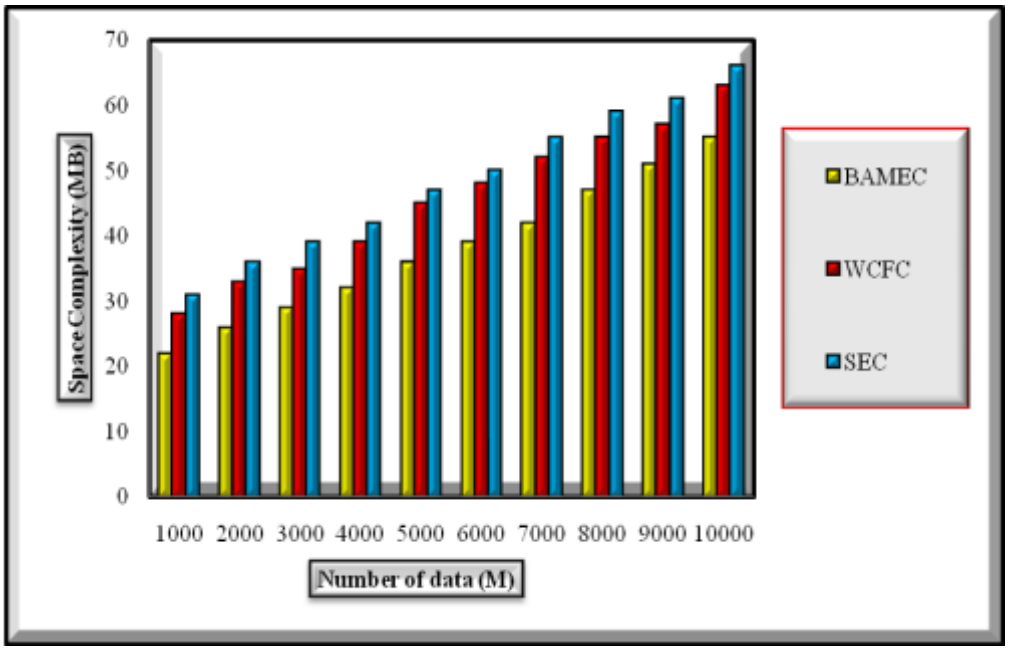

Figure 7 Performance Result of Space Complexity Vs Number of Data

With the help of bootstrap aggregation process, BAMEC technique exactly groups only a related data together in different clusters and thereby avoids the extra usage of memory to store dissimilar data. This helps for BAMEC technique to decrease the memory space needed to store the clustered data as compared to other conventional algorithms. As a result, BAMEC technique minimizes the space complexity of big data analytics by 16% when compared to WCFC [1] and 23 % when compared to SEC [2] respectively.

6. CONCLUSION

The BAMEC technique is designed with the aim of improving the clustering efficiency of big data analytics with minimal computational cost. The aim of BAMEC technique is attained with the help of MEM clustering and bootstrap aggregation process. The developed BAMEC technique processes a large size of data through exploiting the parallelism among clusters. This helps for BAMEC technique to get a lower computational cost for efficient big data analysis as compared to

state-of-the-art works. Further, BAMEC technique enhances the ratio of a number of data correctly grouped with help of majority voting results as compared to state-of-the-art works. Besides, BAMEC technique minimizes the unnecessary use of memory to store dissimilar data by correctly clustering only an interrelated data together in diverse clusters as compared to state-of-the-art works. The performance of BAMEC technique is estimated in terms of clustering accuracy, computational cost, and false alarm rate and space complexity and compared with two conventional works. The experimental result illustrates that BAMEC technique provides better performance with an improvement of clustering accuracy and minimization of computational cost to effectively analyze the huge volume of data when compared to state-of-the-art works.

REFERENCES

[1] Minyar Sassi Hidri, Mohamed Al iZoghlami, Rahma Ben Ayed, "Speeding up the large-scale consensus fuzzy clustering for handling Big Data", Fuzzy Sets and Systems, Elsevier, Volume 348, Pages 50-74, October 2018

[2] Hongfu Liu, Junjie Wu, Tongliang Liu, Dacheng Tao, Yun Fu, "Spectral Ensemble Clustering via Weighted K-Means: Theoretical and Practical Evidence", IEEE Transactions on Knowledge and Data Engineering, Volume 29, Issue 5, Pages 1129 – 1143, May 2017

[3]Tahani Alqurashi, Wenjia Wang, "Clustering ensemble method", International Journal of Machine Learning and Cybernetics, Springer, Pages 1–20, 2018

[4] Mohamed Aymen Ben HajKacem, Chiheb-Eddine Ben N'cir, Nadia Essoussi, "One-pass MapReduce-based clustering method for mixed large scale data", Journal of Intelligent Information Systems, Springer, Pages 1–18, July 2017

[5] SaurabhTewari, U.D.Dwivedi, "Ensemble-based big data analytics of lithofacies for automatic development of petroleum reservoirs", Computers & Industrial Engineering, Elsevier, Volume 128, Pages 937-947, February 2019

[6] José María Luna-Romera, Jorge García-Gutiérrez, María Martínez-Ballesteros, José C. Riquelme Santos, "An approach to validity indices for clustering techniques in Big Data", Progress in Artificial Intelligence, Springer, Volume 7, Issue 2, Pages 81–94, June 2018

[7] Jinchao Ji, Tian Bai, Chunguang Zhou, ChaoMa, Zhe Wang, "An improved k-prototype clustering algorithm for mixed numeric and categorical data", Neurocomputing, Elsevier, Volume 120, Pages 590-596, November 2013

[8] Simone A. Ludwig, "MapReduce-based fuzzy c-means clustering algorithm: implementation and scalability", International Journal of Machine Learning and Cybernetics, Springer, Volume 6, Issue 6, Pages 923–934, December 2015

[9] Younghoon Kim, Kyuseok Shim, Min-Soeng Kim, June Sup Lee, "DBCURE-MR: An efficient density-based clustering algorithm for large data using MapReduce", Information Systems, Elsevier, Volume 42, Pages 15-35, June 2014

[10] Wen-Yen Chen, Yangqiu Song, Hongjie Bai, Chih-Jen Lin, Edward Y. Chang, "Parallel Spectral Clustering in Distributed Systems", IEEE Transactions on Pattern Analysis and Machine Intelligence, Volume 33, Issue 3, Pages 568 – 586, March 2011

[11] Qiuhui Tong, XiuLiBo Yuan, "A highly scalable clustering scheme using boundary information", Pattern Recognition Letters, Elsevier, Volume 89, Pages 1-7, April 2017

[12] Emad A Mohammed, Behrouz H Far, Christopher Naugler, "Applications of the MapReduce programming framework to clinical big data analysis: current landscape and future trends", BioData Mining, Springer, Volume 7, Issue 22, Pages 1-24, December 2014

[13] Weizhong Yan, Umang Brahmakshatriya, Ya Xue, Mark Gilder, Bowden Wise, "p-PIC: Parallel power iteration clustering for big data", Journal of Parallel and Distributed Computing, Elsevier, Volume 73, Issue 3, Pages 352-359, March 2013

[14] Shadi Abpeykar, Mehdi Ghatee, Hadi Zare, "Ensemble decision forest of RBF networks via hybrid feature clustering approach for high-dimensional data classification", Computational Statistics & Data Analysis, Elsevier, Volume 131, Pages 12-36, March 2019
[15] Saeed Shahrivari, Saeed Jalili, "Single-pass and linear-time k-means clustering based on MapReduce", Information Systems, Elsevier, Volume 60, Pages 1-12, August–September 2016
[16] De-Gang, Xu Pan-Lei, Zhao Chun-Hua, Yang Wei-Hua, Gui Jian-Jun He, "A Novel Minkowski-distance-based Consensus Clustering Algorithm", International Journal of Automation and Computing, Volume 14, Issue 1, Pages 33-44, February 2017
[17] Safa Bettoumi, Chiraz Jlassi, Najet Arous, "Collaborative multi-view K-means clustering", Soft Computing, Springer, Volume 23, Issue 3, Pages 937–945, February 2019
[18] Zakaria Benmounah, Souham Meshoul, Mohamed Batouche, Pietro Lio, "Parallel swarm intelligence strategies for large-scale clustering based on MapReduce with application to epigenetics of aging", Applied Soft Computing, Elsevier, Volume 69, Pages 771-783, August 2018
[19] Pengcheng Wei, Fangcheng He, Li Li, Chuanfu Shang, Jing Li, "Research on large data set clustering method based on MapReduce", Neural Computing and Applications, Pages 1–7, 2018
[20] Joaquín Pérez-Ortega, Nelva Nely Almanza-Ortega, David Romero, "Balancing effort and benefit of K-means clustering algorithms in Big Data realms", PLoS ONE, Volume 13, Issue 9, Pages 1-19, 2018
[21]Brazilian E-Commerce Public Dataset:https://www.kaggle.com/olistbr/brazilian-ecommerce/version/7