

International Journal of Engineering in Computer Science



E-ISSN: 2663-3590

P-ISSN: 2663-3582

IJECS 2020; 2(1): 25-31

Received: 12-11-2019

Accepted: 15-12-2019

Mangali Lakshmi Prasanna
Department of Computer
Science, Sri Venkateswara
University, Tirupati, India

Density based clustering and fuzzy clustering for efficient clustering of big data in hadoop ecosystem

Mangali Lakshmi Prasanna

DOI: <https://doi.org/10.33545/26633582.2020.v2.i1.a.29>

Abstract

In this paper suggesting to use parallel distributed Hadoop Map Reduce technology. Map Reduce is a parallel technology which can process bigdata by creating multiple instances of thread. Map will take input data and split it into multiple chunks or parts and distribute all parts to different reducers. All reducers will process the data and send result back to mapper. Mapper gather output from all mappers and then generate a single output. Due to multiple parallel processing of Map Reduce technology allow us to process any amount of data.

In this paper author describing clustering algorithms such Density Based Clustering and Fuzzy Clustering. Both algorithms are not efficient to group all similar data to single cluster and make some data to compromise by putting little un-similar to different clusters. To implement this project author is using National Climatic Data Center (NCDC) dataset which contains climate information. To find out similar temperature on different dates author is applying Hybrid clustering algorithm. From this dataset author is using date and temperature value and then passing date as key to Mapper and temperature as value to Mapper. Mapper always read data in the form of key value pairs. In dataset date we can find at position 6 to14 and temperature we can find at position 39 to 45. Below are some dataset example values.

Keywords: parallel technology, multiple instances, fuzzy clustering

Introduction

Big data as of now producing a buzz in the market and information is quickly developing from being estimated in gigabytes to terabytes, petabytes, and zettabytes ^[1]. Huge information has such huge information prerequisites that applications that were recently used to store and procedure information-Database Management System (DBMS), Relational Database Management System (RDBMS), and so forth are presently bombing the information demand ^[2]. Huge information incorporates incredibly enormous datasets, implying that it isn't workable for usually utilized programming apparatuses to oversee and process that information inside the necessary time frame ^[3]. Along these lines, greatly equal programming stumbling into numerous servers is presently required to deal with this workload ^[4]. Large information expects strategies to uncover bits of knowledge from datasets that are various, complex, and of an enormous scope. A portion of the difficulties of large information handling remember challenges for information catch, addressing the requirement for speed, tending to information quality, managing exceptions, sharing of enormous information, and huge information analysis ^[5]. Various strategies have been proposed to information so as to deal with enormous information datasets, e.g., AI, affiliation systems, bolster vector machines, and clustering ^[6]. Right now, propose another cross-breed grouping strategy to deal with large information.

Literature Survey

A. K-Means Clustering

The diverse research papers using the K-implies grouping for the enormous information are extravagantly talked about beneath, Sreedhar C. *et al.* ^[4] proposed a K-Means Hadoop MapReduce (KM-HMR) for the successful large information bunching. Right now, were introduced for MapReduce (MR) system-based bunching. The KMHMR was the primary strategy, which focused on the use of MR on standard K-implies. The subsequent strategy was to improve the groups quality by limiting the between bunch removes and expanding

Corresponding Author:
Mangali Lakshmi Prasanna
Department of Computer
Science, Sri Venkateswara
University, Tirupati, India

intra-bunch separations.

The proposed KM-HMR

^[5] prescribed a changed Kmeans grouping calculation, which chooses the K-ideal information focuses in the dataset. The principle bit of leeway of choosing the information focuses from the enormous datasets is to forestall exception focuses from including in the last assessment of the group. Increasingly steady outcomes were achieved when the underlying focuses were arranged for proper datasets. Ankita S. also, Prasanta K. Jana ^[6] proposed a K-implies grouping calculation executed in Spark. The proposed K-Means calculation tackled the goals issues which is available in the normal K-Means bunching calculation by earlier robotization of the info groups. It brought about better execution of the Spark structure-based K-implies bunching calculation even with the expanded size of the information and even machine check.

B. Variant of K-means clustering

The distinctive research papers using the Variant of Kmeans bunching for the huge information are intricately talked about beneath, Mohamed Aymen Ben HajKacem *et al.* ^[7] proposed the Accelerated MapReduce-based K-Prototypes (AMRKP) grouping procedure for taking care of large information. Right now perusing and composing of information were done just once on account of this the quantity of Input and Output (I/O) tasks were decreased radically. Furthermost, the proposed plot is reliant on pruning system to quicken the way toward bunching by minimization the excess separation between the inside and information purposes of the group. The created AMRKP outperform the other grouping plans regarding productivity and adaptability. M. Omair Shafiq and Eric Torunski ^[8] proposed an equal K-Medoids grouping calculation dependent on MR structure for doing the compelling bunching of enormous database. The contrived bunching technique was proficient, straightforward and equipped for dealing with the datasets with fluctuating differing attributes, similar to volume, speed and assortment. The reenactment result showed the capacity and attainability of proposed bunching technique in taking care of enormous scope datasets. Mohamed Aymen Ben Haj Kacem *et al.* ^[9] proposed a MR system utilizing K-Prototypes (MR-KP) bunching plan for the powerful information grouping utilizing parallelization. It brought about being a popular and viable bunching technique for blended gigantic datasets. The reenactment was performed on a huge number of tests and the result was exact and versatile in any event, when the size of information is expanded.

C. Fuzzy C-means (FCM) clustering

The diverse research papers using the FCM bunching for the huge information are intricately examined beneath, Simone A Ludwig ^[10] explored the adaptability and parallelization of FCM grouping calculation. The FCM bunching calculation was parallelized utilizing MR structure by illustrating the methodology of guide and lessen work. The approval investigation of the MR-FCM bunching calculation was made to show the adequacy of the proposed calculation as for the part of virtue. Minyar Sassi Hidri *et al.* ^[11] proposed an upgraded FCM grouping calculation utilizing inspecting blended in with part and consolidation

approaches results have outflanked the productivity of other grouping techniques regarding execution time Nadeem Akthar *et al.*

procedure for bunching large information. Starting advance is to part information into unmistakable subsets and work the individual hubs in parallelly. At that point, the subsets were inspected, which were again part haphazardly into particular subsamples. This calculation performed adequately with the furnished assets with upgraded existence complexities.

D. Possibilistic C-implies (PCM) grouping

The diverse research papers using the PCM bunching for the large information are intricately talked about underneath, Qingchen Zhang and Zhikui Chen ^[12] recommended a weighted piece PCM calculation (wkPCM) for grouping the information objects into the reasonable gatherings. The part loads were coordinated for characterizing the item's significance in the bit grouping for limiting the defilement created by the uproarious information. The proposed dispersed wkPCM grouping calculation depended on the MR system which can prepare convincing computational speed for continuous informational collections. Qingchen Zhang *et al.* ^[13] proposed a Privacy Preserving High-Order PCM (PPHOPCM) grouping calculation for playing out the bunching of huge information through the upgrading the goal work. The disseminated HOPCM technique depends on the MR structure with the point of managing large information. In the end, the PPHOPCM was utilized for ensuring the information on cloud by applying the Brakerski-Gentry-Vaikuntanathan (BGV) encryption plot on HOPCM. In the PPHOPCM, participation framework and group focuses were refreshed utilizing polynomial capacities. The formulated PPHOPCM calculation adequately grouped the gigantic dataset and secure the private cloud information.

E. Collaborative Filtering (CF) based bunching

The diverse research papers using CF bunching for the huge information are extravagantly examined underneath, Rong Hu *et al.* ^[14] planned a Clustering-based Collaborative Filtering (ClubCF) approach whose intension is to offer comparative types of assistance enlistment in similar groups for the suggestion of administrations synergistic. This methodology of grouping is included two stages. In the main stage the informational indexes are disintegrated into little pieces of bunches to make them reasonable for next preparing. In the following stage CF is applied to the decided groups. Since the number administrations include in the bunch was not exactly the accessible web benefits the time intricacy of CF was lesser nearly. Subramaniaswamy V. *et al.* ^[15] proposed the prescient component-based CF technique for the successful handling of enormous scope information parallelly. The MR system is utilized for doing the accumulation, sifting and upkeep of the proficient stockpiling. The CF was accustomed to refining of information. The created grouping plan was improved by preparing the information into emoji and tokens through the utilization of notion investigation. The reenactment brought about critical improvement in the multifaceted nature investigation execution.

Clustering analysis techniques

Cluster analysis is a data mining task that aims to provide

for search, recommendation, and organization of data. In clustering techniques, datasets are grouped into a number of clusters each with different attributes^[7, 8]. Clustering is in a class of unsupervised learning techniques, unlike classification, in which similar objects of the dataset are grouped into clusters^[9], and thus form different clusters such that objects in the same cluster groups are very different from each other and objects in the same group or cluster are very similar to each other^[10, 11]. The clusters are known only after the complete execution of the clustering algorithm^[12]. Two clustering algorithms that are used for managing large datasets are Density Based and Fuzzy clustering algorithm each of which is summarized below.

First, we are finding clusters using Density Based algorithm and then applying Fuzzy clustering with MapReduce on density-based cluster to find compromise points and put them in similar cluster.

Density based clustering giving 30% precision and after

applying hybrid algorithm with density and fuzzy giving 100% precision result. Lower the precision value lower the quality of cluster and higher the precision higher the quality of cluster.

Precision can be calculated as: $\text{Total_clusters_find_out} / \text{expected clusters}$. Suppose expected cluster 4 and application finding 10 clusters then precision will be $4 / 10 = 0.4$.

Density Based Clustering: The DBSCAN algorithm is based on this intuitive notion of “clusters” and “noise”. The key idea is that for each point of a cluster, the neighborhood of a given radius has to contain at least a minimum number of points.

Fuzzy Clustering: Fuzzy clustering (also referred to as soft clustering or soft k-means) is a form of clustering in which each data point can belong to more than one cluster.

Results and discussions

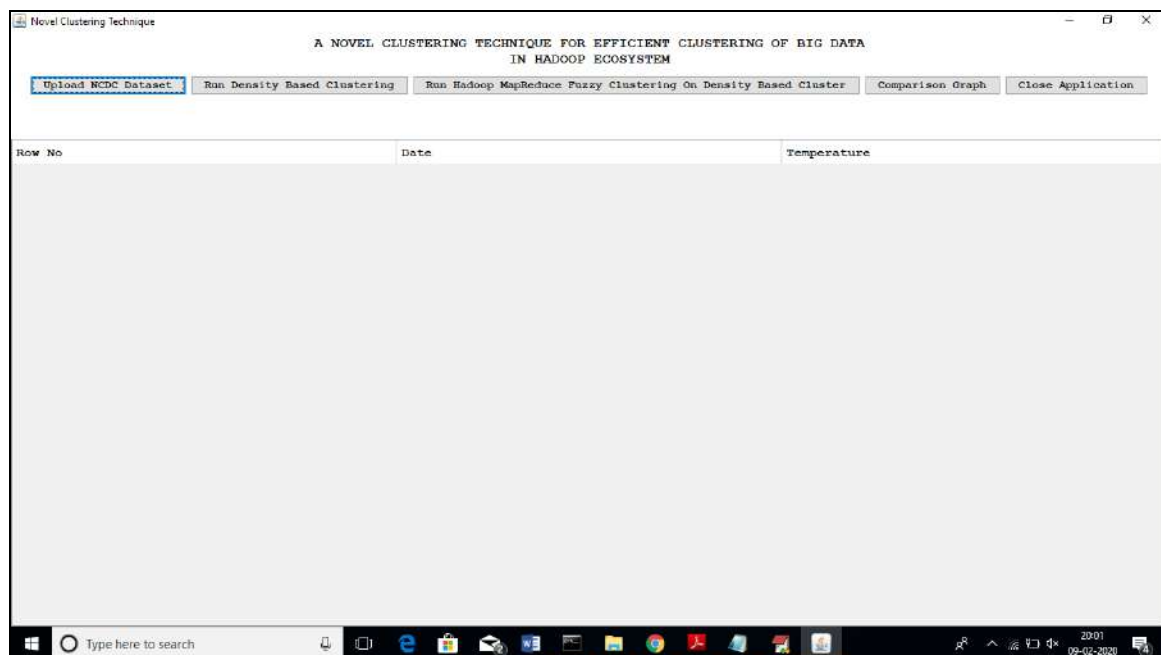


Fig 1: In above screen click on ‘Upload NCDC Dataset’ button and upload dataset

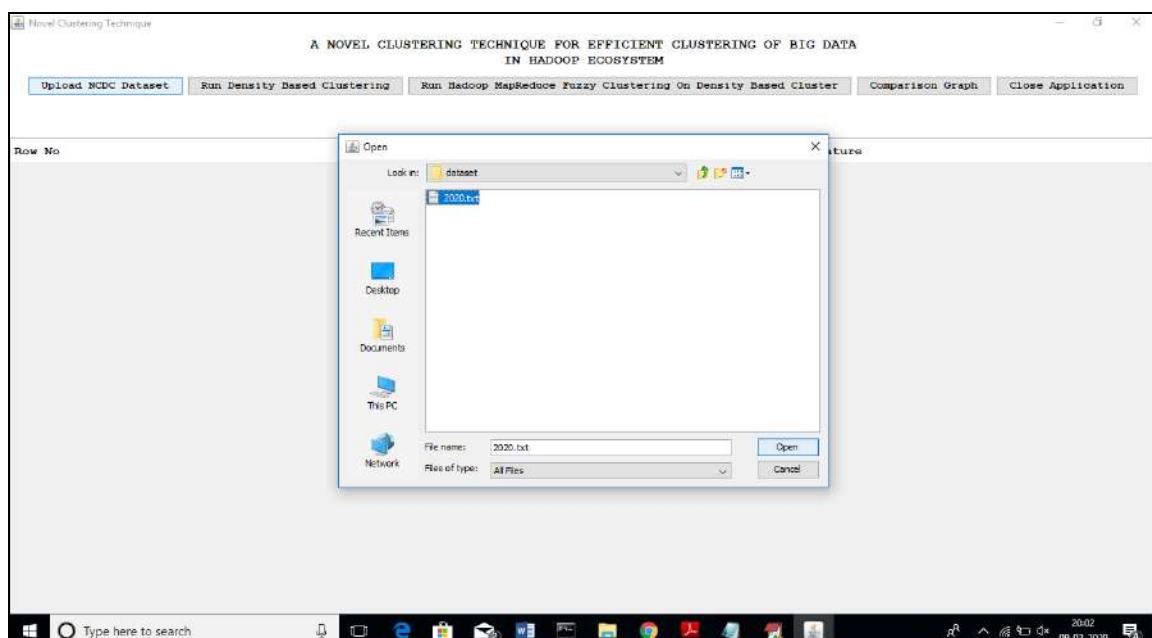
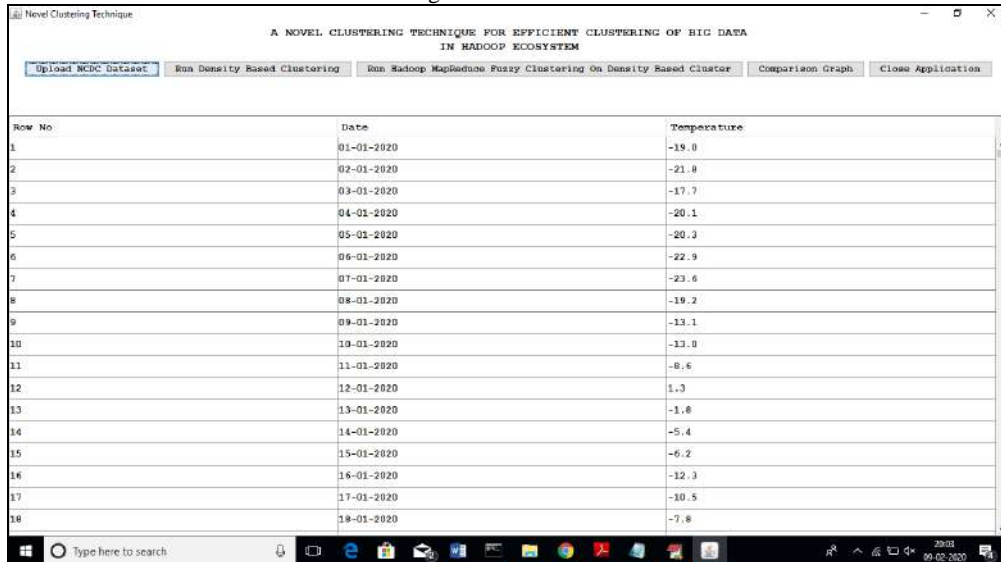
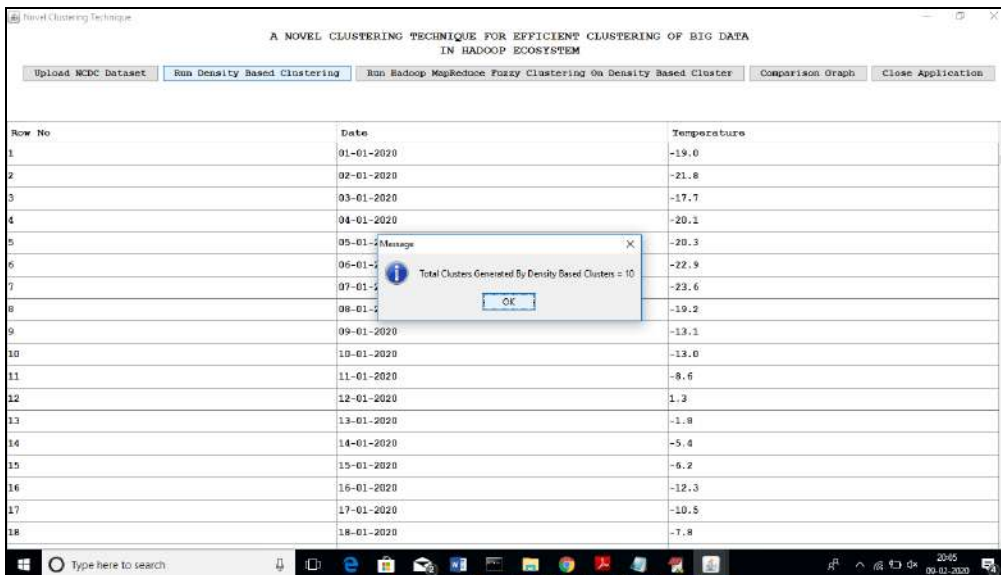


Fig 2: In above screen I am uploading 2020-year dataset of different cities contains temperature and date values. After uploading dataset will get below screen



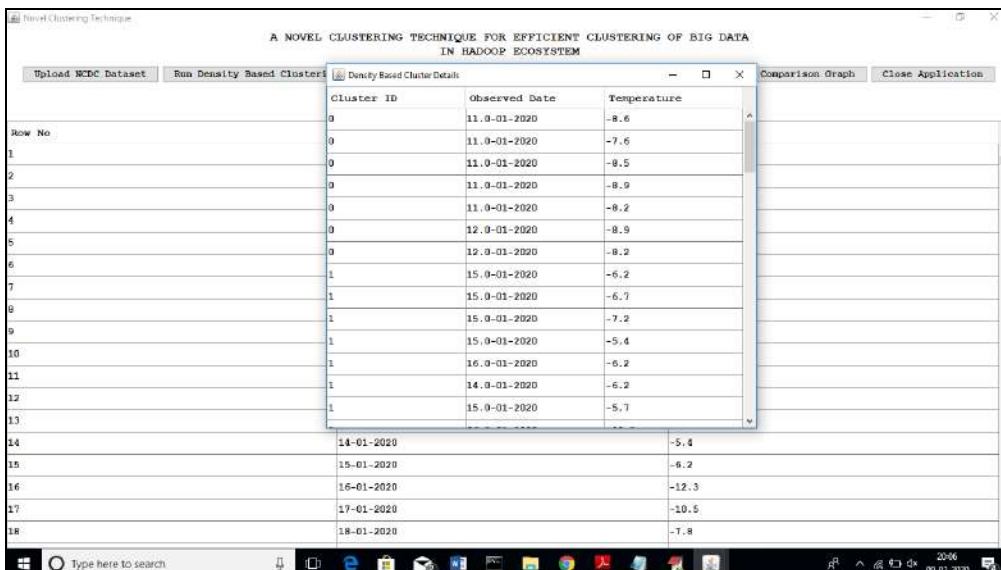
Row No.	Date	Temperature
1	01-01-2020	-19.0
2	02-01-2020	-21.8
3	03-01-2020	-17.7
4	04-01-2020	-20.1
5	05-01-2020	-20.3
6	06-01-2020	-22.9
7	07-01-2020	-23.6
8	08-01-2020	-19.2
9	09-01-2020	-13.1
10	10-01-2020	-13.0
11	11-01-2020	-8.6
12	12-01-2020	1.3
13	13-01-2020	-1.8
14	14-01-2020	-5.4
15	15-01-2020	-6.2
16	16-01-2020	-12.3
17	17-01-2020	-10.5
18	18-01-2020	-7.8

Fig 3: In above screen displaying date and temperature values and now cluster this values by clicking on 'Run Density Based Clustering' button. After clustering will get below screen



Row No.	Date	Temperature
1	01-01-2020	-19.0
2	02-01-2020	-21.8
3	03-01-2020	-17.7
4	04-01-2020	-20.1
5	05-01-2020	-20.3
6	06-01-2020	-22.9
7	07-01-2020	-23.6
8	08-01-2020	-19.2
9	09-01-2020	-13.1
10	10-01-2020	-13.0
11	11-01-2020	-8.6
12	12-01-2020	1.3
13	13-01-2020	-1.8
14	14-01-2020	-5.4
15	15-01-2020	-6.2
16	16-01-2020	-12.3
17	17-01-2020	-10.5
18	18-01-2020	-7.8

Fig 4: In above screen message we can see density based created to total 10 clusters to group all data. Now click on 'ok' button to get all cluster id and their values



Cluster ID	Observed Date	Temperature
0	11.0-01-2020	-8.6
0	11.0-01-2020	-7.6
0	11.0-01-2020	-8.5
0	11.0-01-2020	-8.9
0	11.0-01-2020	-8.2
0	12.0-01-2020	-8.9
0	12.0-01-2020	-8.2
1	15.0-01-2020	-6.2
1	15.0-01-2020	-6.7
1	15.0-01-2020	-7.2
1	15.0-01-2020	-5.4
1	16.0-01-2020	-6.2
1	14.0-01-2020	-6.2
1	15.0-01-2020	-5.7
1	14-01-2020	-5.4
1	15-01-2020	-6.2
1	16-01-2020	-12.3
1	17-01-2020	-10.5
1	18-01-2020	-7.8

Fig 5: In above small screen first column contains cluster id and second column contain date and third column contain temperature values. We can see all closer or related temperature value in 0 cluster and now scroll down screen to see all cluster and its values

Row No.	Cluster ID	Observed Date	Temperature
1	8	04.0-02-2020	2.8
2	9	04.0-02-2020	3.7
3	9	04.0-02-2020	4.6
4	9	05.0-02-2020	4.2
5	9	05.0-02-2020	4.5
6	9	06.0-02-2020	4.2
7	9	05.0-02-2020	4.1
8	9	05.0-02-2020	3.2
9	9	05.0-02-2020	4.3
10	10	02.0-01-2020	-17.5
11	10	02.0-01-2020	-18.0
12	10	02.0-01-2020	-17.1
13	10	01.0-01-2020	-17.5
14	10	02.0-01-2020	-16.9
15	14-01-2020		-5.4
16	15-01-2020		-6.2
17	16-01-2020		-12.3
18	17-01-2020		-10.5
19	18-01-2020		-7.8

Fig 6: In above screen we can see all 10 clusters and now we know density based put some related closed points in different clusters due to that reason we got 10 clusters. Now to reduce clusters size and to put related data in similar cluster we apply Fuzzy algorithm with MapReduce (also called as Hybrid Algorithm) by clicking on 'Run Hadoop MapReduce Fuzzy Clustering On Density Based Cluster' button. After clicking on that button will get below screen

Row No.	Date	Temperature
1	01-01-2020	-19.0
2	02-01-2020	-21.8
3	03-01-2020	-17.7
4	04-01-2020	-20.1
5		
6		
7		
8		
9	09-01-2020	-13.1
10	10-01-2020	-13.0
11	11-01-2020	-8.6
12	12-01-2020	1.3
13	13-01-2020	-1.8
14	14-01-2020	-5.4
15	15-01-2020	-6.2
16	16-01-2020	-12.3
17	17-01-2020	-10.5
18	18-01-2020	-7.8

Fig 7: In above screen we can see Hadoop MapReduce generate 4 clusters and group all data in to 4 clusters. Now click on 'ok' button to get all clusters details in below screen

Row No.	Cluster ID	Observed Date	Temperature
1	0	20200111.0	-8.6
2	0	20200111.0	-7.6
3	0	20200111.0	-8.5
4	0	20200111.0	-8.9
5	0	20200111.0	-8.2
6	0	20200112.0	-8.9
7	0	20200112.0	-8.2
8	0	20200115.0	-6.2
9	0	20200115.0	-6.7
10	0	20200115.0	-7.2
11	0	20200115.0	-5.4
12	0	20200116.0	-6.2
13	0	20200114.0	-6.2
14	0	20200115.0	-5.7
15	14-01-2020		-5.4
16	15-01-2020		-6.2
17	16-01-2020		-12.3
18	17-01-2020		-10.5
19	18-01-2020		-7.8

Fig 8: In above small screen first column contains cluster id and second column contains date and third column contains temperature value. We can see related or closer data is in cluster 0 only. Scroll down screen to see all 4 clusters

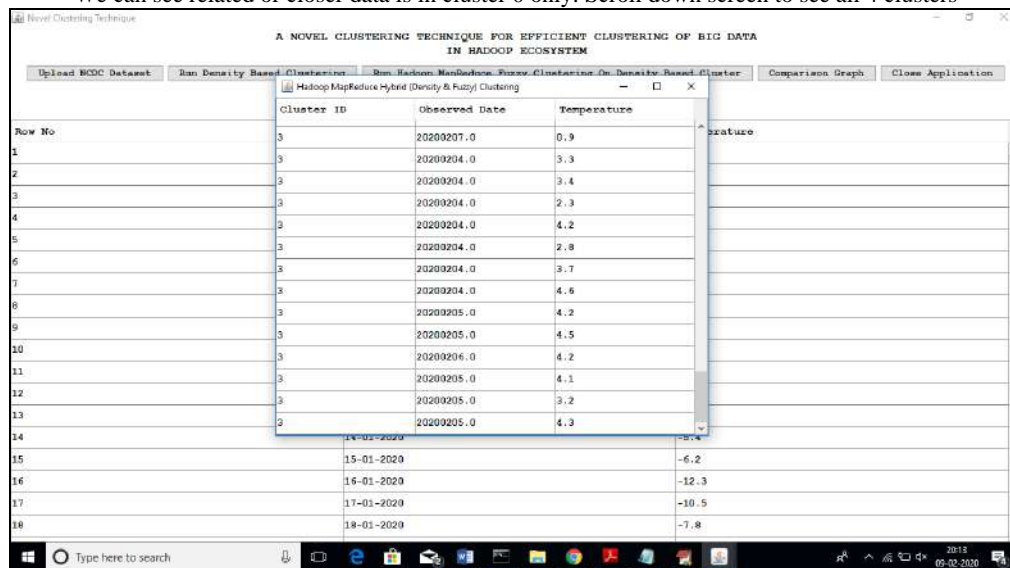


Fig 9: In above screen we can see 3rd cluster also so hybrid MapReduce based generate only 4 clusters from 0 to 3 and this indicates that it put all close similar values goes into appropriate clusters. Now click on 'Comparison Graph' button to see both algorithm precision value in graph

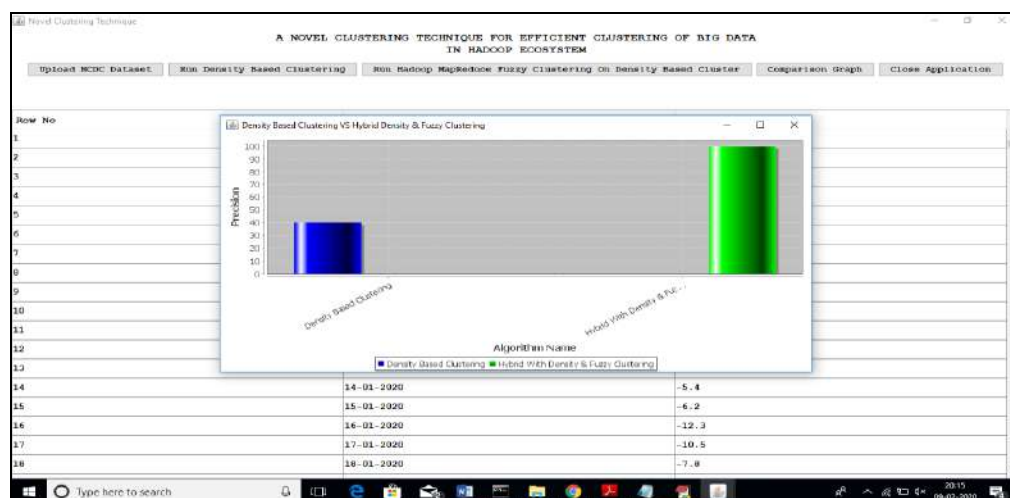


Fig 10: In above graph x-axis represents algorithm name and y-axis represents precision value. From above graph we can see density based got 30% precision value and Hybrid MapReduce based with fuzzy and density-based combination got 100% precision. In below screen we can see MapReduce processing details of data splits

Conclusion

Clustering is a challenging issue that is heavily shaped by data used and problems considered. The proposed algorithms show improvements in terms of execution time. Hadoop can compute map and reduce jobs in parallel to cluster large datasets effectively and efficiently. Suggesting to use parallel distributed Hadoop MapReduce technology. MapReduce is a parallel technology which can process bigdata by creating multiple instances of thread. Map will take input data and split it into multiple chunks or parts and distribute all parts to different reducers. All reducers will process the data and send result back to mapper. Mapper gather output from all mappers and then generate a single output. Due to multiple parallel processing of MapReduce technology allow us to process any amount of data. We have introduced a Density Based and Fuzzy clustering algorithm using Hadoop and to obtain better execution times than those of the standard K-means approach.

Density Based Clustering: The DBSCAN algorithm is based on this intuitive notion of “clusters” and “noise”. The key idea is that for each point of a cluster, the neighborhood of a given radius has to contain at least a minimum number of points.

Fuzzy Clustering: Fuzzy clustering (also referred to as soft clustering or soft k-means) is a form of clustering in which each data point can belong to more than one cluster. here we have used 2020 year of climate dataset and this dataset

References

1. Wullianallur Raghupathi, Viju Raghupathi, "Big data analytics in healthcare: promise and potential", Health information science and systems. 2014; 2(1):3.
2. Bhagyashri S Gandhi, Leena A Deshpande, "The survey on approaches to efficient clustering and classification analysis of big data", International Journal of Engineering Trends and Technology (IJETT). 2016; 36(1):33-39.

3. Ali Seyed Shirkhorshidi, Saeed Aghabozorgi, Teh Ying Wah, Tutut Herawan. "Big Data Clustering: A Review", Computational science and its applications - ICCSA 2014: 14th international conference Guimarães, Portugal, june 30 - july 3, 2014 proceedings, 2016.
4. Chowdam Sreedhar, Nagulapally Kasiviswanath, Pakanti Chenna Reddy. "Clustering large datasets using K-means modified inter and intra clustering (KM-I2C) in Hadoop", Journal of Big Data. 2017; 4(1):27.
5. Nadeem Akthar, Mohd Vasim Ahamad, Shahbaz Khan. "Clustering on Big Data Using Hadoop MapReduce", in proceedings of 2015 IEEE International Conference on Computational Intelligence and Communication Networks (CICN), 2015, 789-795.
6. Ankita Sinha, Prasanta K Jana. "A novel K-means based clustering algorithm for big data", in proceedings of 2016 IEEE International Conference on Advances in Computing, Communications and Informatics (ICACCI), 2016, 1875-1879.
7. 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, 2017, 1-18.
8. M Omair Shafiq, Eric Torunski. "A Parallel K-Medoids Algorithm for Clustering based on MapReduce", in proceedings of 2016 15th IEEE International Conference on Machine Learning and Applications (ICMLA), 2016, 502-507.
9. Mohamed Aymen Ben Haj Kacem, Chiheb-Eddine Ben N'cir, Nadia Essoussi. "MapReduce-based k-prototypes clustering method for big data", in proceedings of 2015 IEEE International Conference on Data Science and Advanced Analytics (DSAA), 2015, 1-7.
10. Simone A Ludwig. "MapReduce-based fuzzy c-means clustering algorithm: implementation and scalability", International Journal of Machine Learning and Cybernetics. 2015; 6(6):923-934.
11. Minyar Sassi Hidri, Mohamed Ali Zoghلامي, Rahma Ben Ayed. "Speeding up the large-scale consensus fuzzy clustering for handling Big Data", Fuzzy Sets and Systems, 2017.
12. Qingchen Zhang, Zhikui Chen. "A weighted kernel possibilistic c-means algorithm based on cloud computing for clustering big data", International Journal of Communication Systems. 2014; 27(9):1378-1391.