Available online at: https://ijact.in

Date of Submission	19/03/2019
Date of Acceptance	29/03/2019
Date of Publication	13/04/2019
Page numbers	3097-3102 (6 Pages)

ISSN:2320-0790

COMPUSOFT
An International Journal of Advanced Computer Technology

OPTIMIZATION OF DBSCAN ALGORITHM USING MAP REDUCE METHOD ON NETWORK TRAFFIC DATA

Hamzah Noori Fejer[1], MohanaedAjmi Falih[2]

Directorate General of Education in Dewanyah
Directorate General of Education in Babylon
hamzah197827@yahoo.com, Mohanaedajmi@gmail.com

Abstract: In this paper, a new method has been proposed to eliminate the weaknesses in the previous algorithms. The proposed method for data density clustering is reduced in the mapping programming model. Our analysis result shows that misleading data was presented to prove the function of the density-based clustering algorithm and the weakness of the base method on them has been represented. Then, local clustering was tested by competing methods for standard data clustering and its superiority to these methods was determined. When passing local clustering to distributed clustering, misleading data was again used to prove the quality of clustering. Distributed clustering quality is lower than local clustering, but it is still superior to the base method. The quality of clustering of the proposed method on competing methods was clearly determined by distributed network clustering. Finally, the method of choosing this parameter was described by evaluating the homogeneity and completeness criteria and the effect of the flexible parameter on different types of data.

Keywords: DBSCAN algorithm, MapReduce method, Network traffic data.

I. INTRODUCTION

The use of reliable protocols, proper data analysis algorithms, and accurate data mining tools and methods are important in this regard given the fact that the discovery of important information among the very enormous collections of non-structured data collected from different sources is heterogeneous and time-consuming. Data clustering is a technique known in various fields and related fields of computer science. The purpose of data clustering is to organize a set of objects in several clusters with similar characteristics.

Usually, different clustering algorithms group data differently. Some of the algorithms are capable to discover proper clustering of data only when the number of the clusters is known. Other algorithms are capable to discover clusters only of particular shapes. There are algorithms that are unable to identify noise data. The DBSCAN algorithm (Density Based Spatial Clustering of Applications with Noise) [1] is recognized as a high quality scalable algorithm for clustering, which is free of these limitations. It belongs to the class of density-based algorithms.

He YB et al. presented a scalable DBSCAN algorithm using MapReduce to remove three major drawbacks in the existing parallel DBSCAN algorithms [2]. Kim et al. proposed a new density-based clustering algorithm which is robust to find clusters with varying densities and suitable for parallelizing the algorithm with MapReduce. Yu et al. proposed an efficient distributed density-based clustering Cludoop algorithm for big data using Hadoop [3]. However, the above researches do not take into account the adjustment problem of the two parameters (min Pts and Eps) on the performance of clustering algorithm.

In this research, we have tried to introduce a new method that eliminates the weaknesses of the previous algorithms, since the clustering algorithms proposed for large data each have a number of disadvantages. The proposed method for data density clustering is reduced in the mapping programming model. This method involves the steps of dividing the data space, local clustering, integration of results, and final labeling, all of which have been presented in reduction mapping programming model.

II. DBSCAN ALGORITHM

The density-based clustering is called the DBSCAN (Density-based spatial clustering of applications with noise) algorithm. This algorithm was presented in 1996 by Esther and his colleagues [1].The purpose of this algorithm is to identify clusters with an arbitrary shape in a noisy environment, while OPTICS is an extension of DBSCAN for different local densities, Another mathematical approach that seems logical is to consider a random variable equivalent to the distance of each sample from its nearest neighbor and calculate its probability distribution. The goal is to identify these scales. Such an approach without parameters has been used in DBCLASD algorithm [4]. DBSLASD considers a cluster as an infinite form in the subset of the data that hopes to distribute the distance to the nearest neighbor and have a previous connection assuming that the points inside each cluster are distributed uniformly (which may or may not be true).

III. RELATED WORKS

In 2007, Liu et al. [5] presented the way to identify different density clusters. The VDBSCA approach is based on a concept called k-dist or a neighboring k-distance. First, the k-dist charts are plotted for k. Then the different values of the selected eps are respectively used in the original DBSCAN algorithm given the failures in this graph.

In 2013, Ting et al. [6] proposed a method called H-Density to solve the problem of identifying various dense clusters. At H-Density, at first two concepts of central cluster and cluster are introduced that relate to two stages of the algorithm implementation. Central clusters are around the city's points, and clusters are composed of central clusters, and the result of clustering will be the same clusters.

In 2012, Esfandany et al. [7] introduced a method called GDCLU to identify clusters of different densities. In GDCLU, the data space is first divided into a tour. Then the number of points in each part is considered as the density of that part. The average density of a part depends on the density of its neighboring parts. The GDCLU defines the average mean value of the density of a part with respect to the average density of its parts, after determining the mean density of a part; that is, the averages of the density are also calculated. After the GDCLU algorithm combines neighboring regions in space whose density is similar to each other by defining the mean variance of the mean of grade density. Measuring this similarity depends on the mean density, average mean density and its variance. Although the results of the work of Esfandany [7] can be promising, it should be noted that by changing the look at the density-based clustering and GDCLU-style space sharing, some details go away using the algorithm and it cannot be expected that clusters of any shape in space will be identified.

In recent years, efforts have been made to cluster large data using a reduction mapping model. Methods of MR-DBSCAN and DBSCAN-MR [2], [8] are studies that focus on parallel implementation of the DBSCAN algorithm using a mapping programming model. The point of subscription of these methods is that it sends adjacent data to a processing node as much as possible.

In 2014, Kim et al. [3] have presented the DBSCAN-MR method for density-based clustering of massive data. In the DBCURE local clustering method, the neighborhood of the samples is calculated elliptically using a multivariate Gaussian function, and basically the concepts contained in the DBSCAN method, such as density and density-related accessibility have defined. In this multivariate Gaussian function, a co-variance matrix has been sampled. To calculate the co-variance matrix representing the distribution of specimens around a sample, the space around each sample is broken into a grid network, and the distance between the sample and its neighbor is several cells, the weighted interval mean of the samples is calculated, and the co-variance matrix is constructed.

Division of data space

The cluster cost analysis parameters have used in the MR-DBSCAN will have a different role, depending on how the reduction mapping model is applied. That the clustering algorithm is performed in the mapping function or the reduction function has an effect on the number of accesses to the disk. In addition, in the proposed method, instead of queries with a specific radius around a sample, which is called the region query, only the closest neighbors are queried. This makes the query simpler and can accelerate the local clustering process.

IV. LOCAL CLUSTERING

The main difference between the flexible algorithms presented in this study with DBSCAN is in expansion stage of the clusters. In the DBSCAN algorithm, the central sample is a sample that has at least minPts points in its neighborhood epsilon; however, in the proposed algorithm, the central sample is a sample whose normal density difference with its average neighborhoods is less than a certain limit (f) That is, an example of which the condition of formula (1) holds true is a central sample.

$$\frac{\left\| Density\,(i) - \frac{\sum_{j\ in\ knnbrs(i)} density\,(j)}{k} \right\|}{\max\left(Density(i), \frac{\sum_{j\ in\ knnbrs\,(i} density\,(j)}{k}\right.} < f$$

In this formula, (Density)i is the density of the i-th sample knnbrs(i) i represents k the closest neighbor of the sample i, k is the input parameter of the algorithm and f is the flexibility parameter.

V. INTEGRATING THE RESULTS

A list of candidate clusters merge is kept to merge clusters. Each element in the list is a collection of label local clusters that must be merged. After completion of the merger phase, the number of merging of this list and the number of public clusters has been given.

VI. ULTIMATE LABEL

To change the data label, it is only enough to replace the label with the mapping from the integration phase. Thus, one final stage can be finalized. In the function of mapping each instance of the received input, the local tag is extracted, according to the general mapping, the public label of the sample is specified and the sample is sent to the output with its public label. The result of this step is sent as output. The result of this step will be saved as the final output.

VII. PROPOSED METHOD EVALUATION

In proposed method evaluation, local clustering is first evaluated with similar methods for the dummy data set, then using the clustering results evaluation criteria, the clustering results of the UCI data set will be compared with similar methods. In order to evaluate the clustering of huge data, external evaluation criteria have been used to evaluate the results and the effect of the parameters on it. In order to evaluate the efficiency of the proposed method, the effect of variables on the number of sub-sections formed in space will be investigated.

Table 1. Summary of KDDCup99 subset information

Dataset Name	Number of attributes	Number of clusters	number of samples
KDDCup99	31	2	100000

Evaluation criteria

Different criteria are used to assess the clustering methods, which can be categorized in the internal and external divisions:

Internal criteria: These criteria rely on clustering results and the distance between different clusters and their samples.

External Criteria: In this clustering criterion, clustering results are compared with the main data labels.

Modified randomized index

An adjusted random variable is a function that shows the similarity between two different modes of assignment of labels with a number in the interval [1 and -1]. The following formula will be used to calculate a simple random index to evaluate the results of clustering of data which actual label is represented by C and its clustered result with K.

$$\boldsymbol{RI} = \frac{\boldsymbol{a+b}}{\frac{1}{2}(\boldsymbol{n-1})\boldsymbol{n}} \tag{2}$$

In this formula, there are a number of pairs of samples in both sets of C and K in a cluster. Also, the number of pairs of samples in both sets C and K in different clusters is shown with b. With regard to formula (2), the adjusted random variable is calculated according to formula (3).

$$\boldsymbol{ARI} = \frac{\boldsymbol{RI - E[RI]}}{\boldsymbol{\max(RI) - E[RI]}} \tag{3}$$

Adjusted countermeasures

The concept of entropy is used to compute the countermeasure criterion. Entropy indicates the amount of uncertainty. The U-tagging entropy is calculated using formula (4).

$$\boldsymbol{H(U)} = \sum_{i=1}^{|v|} \boldsymbol{P(i)\log(P(i))} \tag{4}$$

$$\boldsymbol{P(i) = |Ui|/N}$$

In this formula, it is likely that a random sample of U belongs to the U set.

The mutual information of the two labeling V and U is calculated according to formula (5).

$$\boldsymbol{MI(U,V)} = \sum_{i=1}^{|u|} \sum_{j=1}^{|v|} \boldsymbol{p(i,j)\log\left(\frac{P(i,j)}{P(i)P(j)}\right)} \tag{5}$$

$$P(i,j) = {|Ui \cap Vj|}/{N}$$

In this formula, it is likely that a random sample will be placed in both sets i and j. Formula (6) is used to balance this criterion.

$$AMI = \frac{MI - E[MI]}{\max(H(V), H(V)) - E[MI]} \qquad (6)$$

The calculation method [E] MI is given in [9] Homogeneity, completeness and privilege v.

The two criteria of homogeneity and completeness [10] are complementary. The homogeneity criterion evaluates that each cluster as the final result represents only a real cluster of data. It also evaluates the completeness criterion that all members of a real cluster of data are attributed to a cluster. Score v the harmonic mean of these two criteria. Ideal clustering, at the same time, has the criteria of homogeneity and completeness, and the score v represents the same.

Homogeneity and completeness are calculated with formulas (7) and (8), respectively.

$$h = 1 - \frac{H(C|K)}{H(C)} \qquad (7)$$

$$c = 1 - \frac{H(K|C)}{H(K)} \qquad (8)$$

In this formula, conditional entropy of classes is known by the cluster label and is calculated by formula (8).

$$H(C|K) = -\sum_{C=1}^{|C|} \sum_{K=1}^{|K|} \frac{nck}{n} . \log(\frac{nc}{n}) \qquad (9)$$

In this formula, n is the total number of samples and nck is the number of samples of class c, which is placed in the k-cluster.

Class entropy is also calculated using formula (10).

$$H(C) = -\sum_{C=1}^{|C|} \frac{nc}{c} . \log(\frac{nc}{n}) \qquad (10)$$

In this formula, nc is the number of samples belonging to class c.

Introducing the dataset:

The KDDCup99 dataset is provided to detect network penetration. Due to the processing constraints available, part of this data includes 100,000 samples from the training section. In sum, the KDDCup99 training section, in addition to normal traffic, includes 22 different attacks with a multitude of examples. Sample labels have been converted to normal and abnormal. The reason for this, is a huge difference, the samples belong to different attacks. Thus, the results of data clustering can be applied to the application of malformation detection network. Table 1 presents the summary of these data.

VIII. EVALUATION OF DISTRIBUTED NETWORK CLUSTERING

The main purpose of this paper is to distribute clustering and parallel large network data. Selection of clustering and focusing method and the presentation of clustering were based on the density of cells. Identification of clusters with different densities was also the reason for this. The results of the clustering of the network data with the proposed method are presented in Table 2. These results indicate the absolute superiority of the proposed method on DBSCAN (the maximum number of samples per episode for this experiment was 20,000 samples). Comparing the clustering method for massive data using the MR-DBSCAN and BBSCAN-MR methods, these methods will be at best produce the same result as the DBSCAN base algorithm, implementation of the DBSCAN algorithm in the scikit- learning is also used.

Table 2: Distributed clustering results of network data

Data collection	Algorithm	parameters	H	C	V	ARI
KDD99 Subsct	DBSCAN	minPts=10,cps=10	0.35	0.235	0.281	0.187
	MR-KNNCA(VA)	K=20,f=0.3	0.692	0.318	0.436	0.547
	MR-KNNCA(VM)	K=30,f=0.3	0.867	0.434	0.579	0.660
	MR-KNNCA(LA)	K=20,f=0.3	0.673	0.370	0.434	0.539
	MR-KNNCA(LM)	K=30,f=0.3	0.868	0.383	0.532	0.571

Here we can see the effect of variance on the quality of clustering. By choosing attributes based on variance in VM policy, the best quality of clustering is achieved. The reason is that as the number of attributes increases, the richer the information is, the more targeted the decisions will be. To select between two attributes, the probability that the variance and the length of the results are similar is much greater than that of the 41 attributes; therefore, in the second case, the results will have a significant difference.

Table 2 shows the effect of space division policies on the quality of clustering. From the perspective of the cost of clustering distributed the division of samples into more areas of more overhead. The number of space segments for different space division policies is plotted in Fig. 1. As expected, the less data is used in the statistics; the number of space failures will be reduced.

After investigating the quality of clustering, the turning point will investigate the effect of the parameters on clustering. As the number of samples increases, choosing the k parameter is not a big challenge; however, the flexibility parameter still has a significant effect on clustering results.

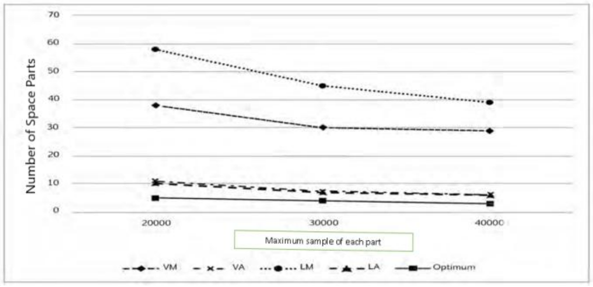

Figure 1: The Effect of Different Policies on Space Divisions

The Effect of Flexibility Parameters on Clustering

The flexibility parameter determines the maximum difference between the estimated densities of a sample with its neighboring samples in a cluster. Whatever the size, the clusters are expected to merge together and reduce their number. In this section, the study of the effect of the flexibility parameter, the two criteria of homogeneity and completeness are investigated.

The proposed behavior will be compared in both local clustering and distributed clustering. For a distributed state, according to the results of Table 2, only the results of the VM policy are reported. This policy has led to significant results for data on crescent and network data.

The effect of the flexibility parameter on the clustering results when working with the firewall data set is shown in Fig. 2. Firstly, by increasing the flexibility parameter, both criteria of homogeneity and completeness increase. When the parameter F reaches to 0.2, the homogeneity will be decreased. At this time, clusters of close together began to merge. As a result, in a cluster there are samples of several different clusters and low homogeneity. In this case, with the addition of clusters close to each other for lower values of F, part of their samples are mistakenly placed in another cluster, the score for completeness increases. If this happens, the fullness parameter will also increase. Indeed, if all specimens are placed in a cluster, completeness will peak to its maximum, but if a small portion of a cluster is separated from the rest of the sample, completeness will also decrease. This is for larger values than 0.6.

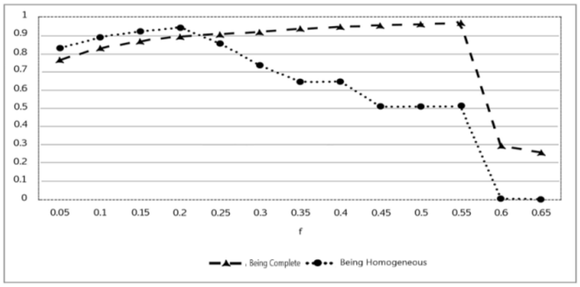

Figure 2: The Effect of the Flexibility Parameter on the Clustering of the Fireworks Dataset

In the case of the crescent data set, the results of the effect of the flexibility parameter on clustering are shown in Fig. 3. The degree of homogeneity, as in the case of the data set of the fireworks, falls after its ascending course. An interesting point is that completeness is still high. That is, in the crescent data set, samples belonging to a cluster rarely fall into different clusters. The reason for this is that the density of the samples in each crescent has a significant degree of uniformity. The greater the flexibility parameter in this clustering, the closer clusters will be merged and the cluster homogeneity will decrease. When the amount of flexibility is low, some of the border samples may be detected as noise, reducing the degree of completeness, but leaving no noise with increasing flexibility. The integration of two distinct clusters in a cluster also has no effect on completeness.

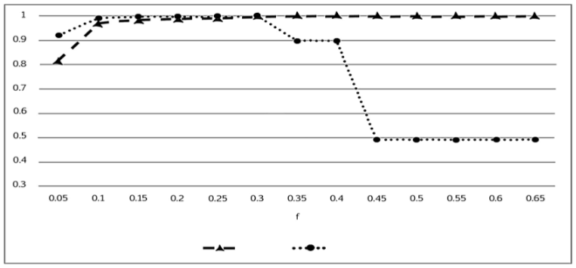

Figure 3: The Effect of the Flexibility Parameter on the Clustering of the Crescent Data Set

The data sets of the network set in this cluster study have a different situation. These data have two labels for the two clusters, but the cluster of maladaptive data is actually composed of several different attacks. It can be expected that this cluster consists of several distinct clusters. Normal data, despite the fact that they are all in the category of anomalies, have different characteristics and can be considered in several independent clusters. In such a dataset, one can expect that never be close to its final value, i.e., one. In the case of cluster interference, if the flexibility parameter increases excessively, it can be expected that the clusters that are close to each other are malformed and merged and reduce homogeneity. Figure 4 is a proof of this claim. The level of homogeneity is similar to the two previous datasets, but the approximate variation of the approximation is negligible. The change in homogeneity is also milder than the previous two datasets. The reason for this can be found in the imbalance between different clusters. Although in general, the number of normal and abnormal samples is close to each other, but the different clusters of these two large clusters have different sizes. While most of the samples do not change hands-on with the flexibility of the hand, changing the smaller clusters and merging them will change the amount of homogeneity. The low number of specimens involved in these changes makes the slope of the changes not great.

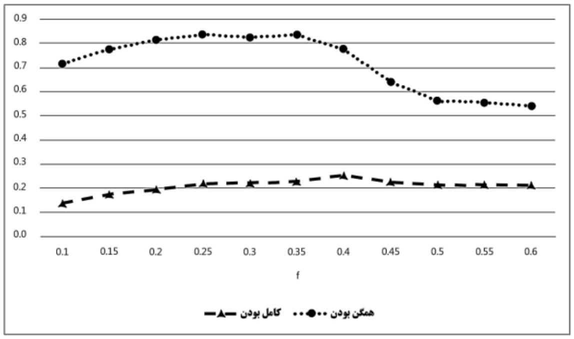

Figure 5: The Effect of the Flexibility Parameter on the Clustering of the Crescent Data Set

In sum, it can be said that in order to fine-tune the flexibility parameter, there should be a complete understanding of how data clusters are distributed in space. The flexibility parameter can be in the interval [1 and 0], but in practice it can be seen that the range [0.6 and 0.4] is the last range in which expected results can be expected; therefore, the problem of setting this parameter without The existence of a cognition of the distribution of data in space is still not very complicated.

IX. Conclusion

The results showed that there might be misleading data presented to prove the function of the density-based clustering algorithm and the weakness of the basic method on them is also presented. Then, local clustering was tested by competing methods for standard data clustering, and its interface with these methods was determined. When passing local clustering to distributed clustering, the misleading data was again used to prove the quality of clustering. Distributed clustering quality is lower than local clustering, but it is still superior to the base method. The quality of clustering of the proposed method on the competing methods was clearly indicated by distributed clustering of network data. Finally, by choosing the criteria for homogeneity and completeness and the effect of the flexible parameter on different types of data, the method of selecting this parameter was also described.

X. References

[1] Ester, M., Kriegel, H.P. et al. (1996) A Density-Based Algorithm for Discovering Clusters in Large Spatial Database with Noise. KDD, 226-231.

[2] He YB, Tan HY, Luo WM, et al. MR-DBSCAN: a scalable MapReduce-based DBSCAN algorithm for heavily skewed data, Front Comput Sci 2014; 8(1): 83–99, DOI 10.1007/s11704-013-3158-3.

[3] Kim Y, Shim K, Kim MS, et al. DBCURE-MR: an efficient density-based clustering algorithm for large data using MapReduce. Inform Syst 2014; 42: 15–35.

[4] Birant, D. and A. Kut (2007). "ST-DBSCAN: An algorithm for clusteringspatial–temporal data." Data & Knowledge Engineering 60(1): 208-221.

[5] Liu, P., et al .)2007(."VDBSCAN: varied density based spatial clustering of applications with noise". Service Systems and Service Management, 2007 InternationalConference on, IEEE.

[6] Ting, K. M., et al. (2013). "DEMass: a new density estimator for big data." Knowledge and information systems 35.493-524 :(3)

[7] Esfandani, G., et al. (2012). "GDCLU: a new Grid-Density based CLUstring algorithm". Software Engineering, Artificial Intelligence, Networking and Parallel & Distributed Computing (SNPD), 2012 13th ACIS International Conference on, IEEE.

[8] He, Y., et al. (2011). "Mr-dbscan: an efficient parallel density-basedclustering algorithm using mapreduce". Parallel and Distributed Systems (ICPADS), 2011 IEEE 17thInternational Conference on, IEEE.

[9] Vinh, N. X., et al. (2009). "Information theoretic measures for clusterings comparison": is a correction for chance necessary? Proceedings of the 26th annual international conference on machine learning, ACM.

[10] Rosenberg, A. and J. Hirschberg (2007). "V-Measure: A Conditional Entropy-Based External Cluster Evaluation Measure". EMNLP-CoNLL.