IOSR Journal of Engineering (IOSRJEN) **www.iosrjen.org**
ISSN (e): 2250-3021, ISSN (p): 2278-8719
Vol. 08, Issue 12 (December. 2018), ||V (V) || PP 33-40

DBSCAN-BRNNDE: A Density-based Clustering Algorithm using Bichromatic Reverse nearest Neighbor Density Estimates

Mr.S.Premkumar[1], Dr.N.Revathy [2]
[1]Ph.D Research Scholar, [1]Department of Computer Science
[2]Associate Professor, [2]Department of Master of Computer Applications,
Hindusthan College of Arts and Science, Coimbatore, India
Corresponding Author: Mr.S.Premkumar

Abstract: In data mining, clustering algorithms is one of mostly used area for many applications. With the growth of the online applications, studies on cluster development were performed consequently as to reduce noise and sustain continuity of grouping results. Several clustering algorithms have been introduced, such as the evolutionary methods and spectral clustering, however none of the works has been were engaged to solving the density-based clustering problem. Recently new clustering algorithm, RNN-DBSCAN, is introduced which uses RNN counts as an estimate of observation density. But still the accuracy is challenging task, so Density-based Clustering Algorithm using Bichromatic Reverse Nearest Neighbor Density Estimates (DBSCAN-BRNNDE) clustering algorithm is introduced this work with two major steps: First, issues of computation complexity is decreased via the use of a single parameter (choice of Bichromatic k nearest neighbors), and second, an enhanced capability for managing huge variations in terms of cluster density (varied density). To measure the results of DBSCAN-BRNNDE, with respect to existing methods, many artificial datasets are used in this work. Results are evaluated in terms of Adjusted Rand Index (ARI) and Normalized Mutual Information (NMI) , it is implemented in MATLAB environment.

Index Terms: Data mining, clustering, Density-based Clustering (DBSCAN), Bichromatic Reverse Nearest Neighbor Density Estimates (BRNNDE), and evaluation metrics.

Date of Submission: 14-12-2018 Date of acceptance: 29-12-2018

I. INTRODUCTION

Clustering is a prominent unsupervised approach for naturally discovering classes, ideas, or gathering of examples. Grouping includes partitioning an arrangement of data into a predefined number of groups. The way toward gathering an arrangement of physical or theoretical data into classes of comparative articles is called grouping [1]. There are numerous kinds of information that regularly happen in group examination, for example, short-term scaled factors, parallel variables, supposed, mixed and proportion factors.

Grouping is a testing field of research in which its potential applications represent their own uncommon prerequisites. Grouping calculations are customarily isolated into one of a few classifications which incorporate partitioning, hierarchical, model, density, and grid based methodologies.

Density-based clustering forms the clusters of densely gathered objects separated by sparse regions; it has the advantage that it can discover the clusters of arbitrary shapes and easily filter out noise objects. DBSCAN [2], OPTICS [3], and DENCLUE [4] are widely used density-based clustering algorithms. OPTICS is an extension to DBSCAN that solves the problem of parameter selection, and DENCLUE 2.0 [4] is an upgrade of DENCLUE [2] that improves its performance.

Several desirable properties of density-based clustering include an ability to handle and identify noise, discover clusters with arbitrary shapes, and automatic discovery of the number of clusters. In contrast to DBSCAN, reverse nearest neighbor approaches such as RECORD [5], IS-DBSCAN [6], ISBDBSCAN [7], and RNN-DBSCAN, define observation density using reverse nearest neighbors. All of these approaches require the use of a single parameter k, number of nearest neighbors, which is also used to identify dense observations. For example, in RECORD, if an observation has k or more reverse nearest neighbors it is identified as a dense observation. Additionally, in RECORD, observation reachability is defined by core observation traversals of the reverse k nearest neighbor graph.

Density-based Clustering Algorithm using Bichromatic Reverse Nearest Neighbor Density Estimates(DBSCAN-BRNNDE) clustering algorithm is introduced this work with two major steps: First, issues of computation complexity is decreased via the use of a single parameter (choice of Bichromatic k nearest neighbors), and second, an enhanced capability for managing huge variations in terms of cluster density (varied density).

II. REVIEW WORK

Hinneburg et al [4] introduced a new hill climbing procedure for Gaussian kernels, which adjusts the step size automatically at no extra costs. Also prove that the procedure converges exactly towards a local maximum by reducing it to a special case of the expectation maximization algorithm. Results show experimentally that the new procedure needs much less iterations and can be accelerated by sampling based methods with sacrificing only a small amount of accuracy.

Brecheisen et al [8] parallelize the density-based clustering algorithm DBSCAN. First, the data is partitioned based on an enumeration calculated by the hierarchical clustering algorithm OPTICS, so that similar objects have adjacent enumeration values. The use the fact that clustering is based on lower-bounding distance values conservatively approximates the exact clustering. By integrating the multi-step query processing paradigm directly into the clustering algorithms, the clustering on the slaves can be carried out very efficiently. Finally, results show that the different result sets computed by the various slaves can effectively and efficiently be merged to a global result by means of cluster connectivity graphs.

Bohm et al [9] propose Compute Unified Device Architecture (CUDA)-DClust, a massively parallel algorithm for density-based clustering for the use of a Graphics Processing Unit (GPU). While the result of this algorithm is guaranteed to be equivalent to that of DBSCAN, also demonstrate a high speed-up, particularly in combination with a novel index structure for use in GPUs.

Louhichi et al [10] present a novel density-based clustering algorithm called MultiDensity ClUsTering (MDCUT). The presented algorithm has the merit of clustering data with varied density. Results show experimentally that the MDCUT algorithm detects correctly the density levels in a data set and succeeds to discover arbitrarily shaped clusters in decreasing density order.

Birant and Kut [11] present a new density-based clustering algorithm, ST-DBSCAN, which is based on DBSCAN. Also propose three marginal extensions to DBSCAN related with the identification of (i) core objects, (ii) noise objects, and (iii) adjacent clusters. In contrast to the existing density-based clustering algorithms, our algorithm has the ability of discovering clusters according to non-spatial, spatial and temporal values of the objects.

Su et al [12] proposed a fast incremental clustering algorithm by changing the radius threshold value dynamically. The algorithm restricts the number of the final clusters and reads the original dataset only once. At the same time an inter-cluster dissimilarity measure taking into account the frequency information of the attribute values is introduced. It can be used for the categorical data. The experimental results on the mushroom dataset show that the proposed algorithm is feasible and effective.

Chen et al [13] proposed a novel parameter free clustering algorithm named as APSCAN. Firstly, utilize the Affinity Propagation (AP) algorithm to detect local densities for a dataset and generate a normalized density list. Secondly, combine the first pair of density parameters with any other pair of density parameters in the normalized density list as input parameters for a proposed DDBSCAN (Double-Density-Based SCAN) to produce a set of clustering results. Thirdly, developed a updated rule for the results obtained by implementing the DDBSCAN with different input parameters and then synthesize these clustering results into a final result.

He et al[14] proposed an efficient parallel density-based clustering algorithm and implement it by a 4-stages MapReduce paradigm. Furthermore, also adopt a quick partitioning strategy for large scale non-indexed data. This work study the metric of merge among bordering partitions and make optimizations on it. At last, evaluate this work on real large scale datasets using Hadoop platform. Results reveal that the speedup and scale up of proposed work are very efficient.

Welton et al[15] introduced new version of DBSCAN, called MRSCAN, which uses a hybrid parallel implementation that combines the Multicast/Reduction Network (MRNet) tree-based distribution network with General-purpose computing on graphics processing units (GPGPU) equipped nodes. MRSCAN avoids the problems of existing implementations by effectively partitioning the point space and by optimizing DBSCAN's computation over dense data regions. Implemented these clustering algorithms on geolocated Twitter dataset and image data obtained from the Sloan Digital Sky Survey. At its largest scale, proposed clustered 6.5 billion points from the Twitter dataset on 8,192 GPU nodes on Cray Titan in 17.3 minutes. All other parallel DBSCAN implementations have only demonstrated the ability to cluster up to 100 million points.

Campello et al[16] proposed a theoretically and practically improved density-based, hierarchical clustering method, providing a clustering hierarchy from which a simplified tree of significant clusters. For obtaining a “flat” partition consisting of only the most significant clusters (possibly corresponding to different density thresholds), proposed a novel cluster stability measure, formalize the problem of maximizing the overall stability of selected clusters, and formulate an algorithm that computes an optimal solution to this problem. Also demonstrate that proposed approach outperforms the current, state-of-the-art, density-based clustering methods on a wide variety of real world data.

III. PROPOSED METHODOLOGY

Density-based Clustering Algorithm using Bichromatic Reverse Nearest Neighbor Density Estimates(DBSCAN-BRNNDE) clustering algorithm is introduced this work with two major steps: First, issues of computation complexity is decreased via the use of a single parameter (choice of Bichromatic k nearest neighbors), and second, an enhanced capability for managing huge variations in terms of cluster density (varied density). To measure the results of DBSCAN-BRNNDE, with respect to existing methods, many artificial and shaped-based datasets are used in this work. Results are evaluated in terms of Adjusted Rand Index (ARI) and Normalized Mutual Information (NMI) , it is implemented in MATLAB environment.

Cluster formation

Let A represent a set of observations of size $n = |A|$, such that each observation in A is drawn from a d-dimensional space of real values,$\forall a \in A: a \in \mathbb{R}^d$. For any two observations $a, b \in A, let\ dist(a, b)$ represent a distance function (metric or non-metric) that returns the distance between observations **a** and **b**. Note that the Euclidean distance was used for all results produced in this work.

$$dist(a,b) = \sqrt{\sum_{i=1}^{d} (b_i - a_i)^2}$$

Reverse nearest neighborhood of observation **a** is defined by the function $R_k(\boldsymbol{a}) = R$ where R satisfies the following conditions:

1. $R \subseteq A/\{a\}$
2. $\forall b \in R: a \in N_k(b)$

Similar to DBSCAN, three types of observations are defined in X which are core, boundary, and noise. An observation $\boldsymbol{a} \in \boldsymbol{A}$ is a core observation $iff\ |R_k(a)| \geq k$ whereas **x** is a boundary or noise observation $if\ |R_k(a)| < k.$

Border observations are those of the aforementioned set which are assigned to clusters, while noise observations remain unclustered.

Next, observation reachability definitions are presented which will be used in defining clusters. An observation's direct reachability is limited to its set of k nearest neighbours along with its status as a core observation.

Directly density-reachable: A observation **a** is directly density reachable from a observation **b** if

1. $a \in N_k(b)$
2. $|R_k(b)| \geq k(\text{core observation condition})$

Directly density reachable is non-symmetric for non-core observations, and not guaranteed to be symmetric in the case of core observations. The latter of the two cases being due to the fact that the nearest neighbor relationship is non-symmetric, and the former as no observation is reachable from a non-core observation.

Density-reachable: A observation a is density reachable from a observation b if there is a chain of observations a_1,..., a_m, a_1 = b, a_m =a such that where $|R_k(a)| \geq\ \ k$

1) $\forall 1 \leq i \leq m-1:\ a_{i+1}$ is directly density-reachable from a_i is directly density-reachable from a_{i+1} and where$|R_k(a)| \leq\ \ k$

1) a_m is directly density-reachable from a_{m-1}
2) $\forall 1 \leq i \leq m-2:\ a_{i+1}$ is directly density-reachable from a_i is directly density-reachable from a_{i+1}. Density reachable is a canonical extension of directly density reachable, and is transitive though not symmetric.

However, this relationship is symmetric where observations a and b are core observation. Given cluster C, let the density of C be the maximum of the directly density-reachable distances among core observations in C.

For all observations, traversed in some arbitrary order, if the current (seed) observation has yet to be assigned to a cluster and is a core observation, then it is assigned to a new cluster. This new cluster is expanded by a breadth first search of all unclustered reachable observations, density-connected, from the seed observation. Let G_{RNN} = (V,E) represent the directed k nearest neighbour graph, such that V is equal to the set of observations, V = X, and $\forall(\boldsymbol{u},\boldsymbol{v})\ \in\ E$ **v** is a k nearest neighbor of **u**, $\boldsymbol{v}\ \in\ N_k(\boldsymbol{u})$. The set of core observations, Core, is defined as observations whose number of reverse nearest neighbors is equal to or greater than k.

Given two samples D (the servers) and R (the data objects), and a server q ∈ D, a bichromatic reverse nearest neighbor (BRNN) query finds the set of objects whose nearest data point is q.

BRNN attracts considerable attention since its first seminal work [17]. To efficiently find the BRNNs for a cluster point, Voronoi polygon is widely used under various circumstances such as static or continuous processing [18-19]. In these works, the Voronoi polygon determines candidate or accurate region for the query point, within which object points are the query point's BRNN result. In this work, the clustering points samples instead of a point to the samples, and the samples returns object points that are BRNNs to every point in the region. The datapoint then refines the actual BRNNs based on his/her actual location point. While this solution still exposes a query region, work supersedes it by revealing no location data of the cluster point.

The data points a BRNN query $q \in D$, for which the server returns the set of objects whose nearest server is q. Formally, $BRNN(q) = \{r \in R \mid \forall s \in D : dist(r, q) \leq dist(r, s)\}$. As for the privacy requirement, the server should not learn any information about q.

An interesting property to consider with respect to DBSCAN-BRNNDE is the effect dataset size n has on the choice of k. In particular, one would like to know if the choice of k is independent of n. This might in fact be assumed as the expected number of BRNNDE, k, is independent with respect to the size of the dataset n. With this in mind, let us assume that the choice of k should not be affected by n, except in cases where n is not sufficiently large enough to represent data.

Performance of DBSCAN-BRNNDE is dependent on the choice of k, two potential heuristics for selecting an appropriate value of k are presented. Of all discussed prior BRNNDE approaches by suggesting two distinct heuristics. One of said heuristics is based on the observation that good clustering solutions occur across multiple values of k. The first approach presented here is based on this observation.

To determine an appropriate value of k, clustering stability is examined with respect to the frequency of number of clusters, $l_k(X)$, for some range of k. Here it is assumed that a correlation exists between clustering performance and the frequency in which a clustering solution, defined by number of clusters, occurs. In particular, it is assumed that good performance can be found at spikes in the histogram of $l_k(X)$ over an appropriate range of k. Note that this only suggests an appropriate number of clusters which may be obtained from some set of k values. BRNNDE is based on the general assumption that in good clustering, observations in a cluster should be tightly connected while observations belonging to different clusters should be far apart.

With respect to tightly connected, for some cluster C_i, $DSC(C_i)$ defines the density sparseness of a cluster as the maximum distance in the minimum spanning tree of the core observations in C_i. With respect to clusters being far apart, for two clusters C_i and C_j , $DSPC(C_i,C_j)$ defines the density separation for a pair of clusters as the minimum distance between core observations in C_i and C_j. For the clustering $C = (C_1,..,C_l)$, the validity of cluster C_i is defined by the cluster validity index $V(C_i)$ which compares the cluster's DSC to its minimum DSPC across the clustering:

$$V_c(C_i) = \frac{\min_{C_j \in C, C_j \neq C_i} DSPC(C_i, C_j) - DSC(C_i)}{\max\left(DSPC(C_i, C_j) - DSC(C_i)\right)}$$

The validity index of clustering C, DBCV (C), is defined as the weighted average of the cluster validity index for all clusters in C.

$$DBCV\ (C) = \sum_{i=1}^{l} \frac{|C_i|}{n} \times V_C(C_i)$$

Note that $DBCV\ (C) \in [-1, +1]$ where higher values indicate a better clustering.

IV. RESULTS AND DISCUSSION

To evaluate the performance of DBSCAN-BRNNDE, with respect to prior approaches, several artificial, datasets from [20] were used. In addition to these, several artificial datasets of varying sizes where generated using the scikit learn toolkit [21], along with a grid-based dataset generated to highlight the density variation limitation of DBSCAN. A summary of each dataset is provided in Table 1.

Table 1. Artificial Datasets

Data	Observation	Classes	Dimensions
Aggregation	788	7	2
D31	3100	31	2
Flame	240	2	2
Jain	373	2	2

With respect to performance, Adjusted Rand Index (ARI) [22] and Normalized Mutual Information (NMI) [23] were used for evaluation. ARI represents the similarity measure between two clusterings that is adjusted for chance and is related to accuracy, while NMI quantifies the amount of information obtained about one clustering, through the other clustering (i.e., the mutual dependence between the two). In the case of observations being identified as noise, each noise observation was treated as a distinct singleton cluster for both ARI and NMI. For ARI results, clustering Purity is also presented which is a weighted average of the percentage of observations belonging to the dominant class in each cluster. Noise observations were ignoring in the calculation of purity.

Table 2 shows the results of various clustering methods such as DBSCAN-BRNNDE (BRNNDE), RNN-DBSCAN (RNN), and REC with respect to different metrics such as ARI, Purity, No. of cluster, and NMI on the artificial datasets. From these results, it concludes that the proposed BRNNDE clustering algorithm. More importantly, in each case BRNNDE was able to identify the underlying classes of each dataset (i.e., at the maximum ARI solution number of clusters equals the number of expected classes), whereas each of the other approaches fail at this task in at least one case

Table2. ARI Performance on Artificial Datasets

Data	Metrics	BRNNDE	RNN	REC
Aggregation	**ARI**	0.99	0.97	0.742
	Clusters	7	7	7
	Purity	0.98	0.96	0.93
	NMI	156	163	168
D31	**ARI**	0.923	0.890	0.63
	Clusters	32	35	39
	Purity	0.979	0.971	0.93
	NMI	165	169	358
Flame	**ARI**	0.982	0.953	0.687
	Clusters	2	2	2
	Purity	0.985	0.9715	0.9326
	NMI	30	22	45
Jain	**ARI**	0.986	0.963	0.453
	Clusters	2	2	2
	Purity	0.98	0.93	0.91
	NMI	25	50	105

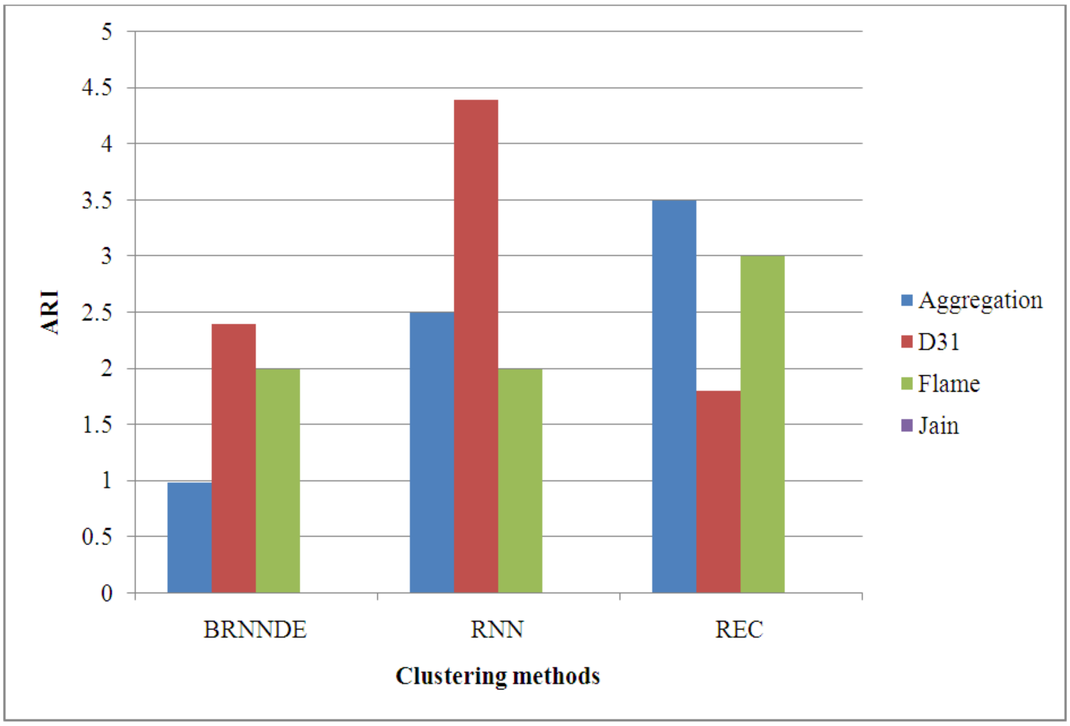

Figure1. ARI Performance on Artificial Datasets with clustering methods

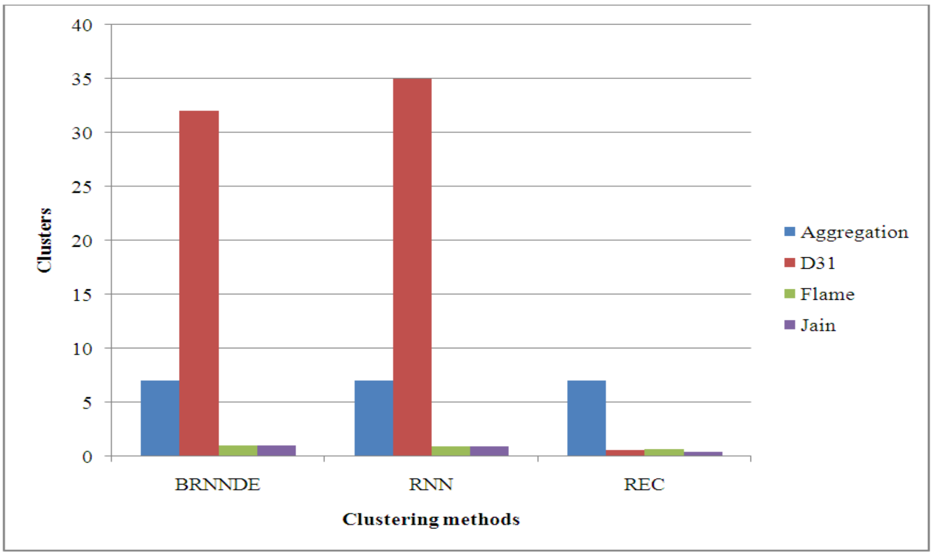

Figure2. Clusters on Artificial Datasets with clustering methods

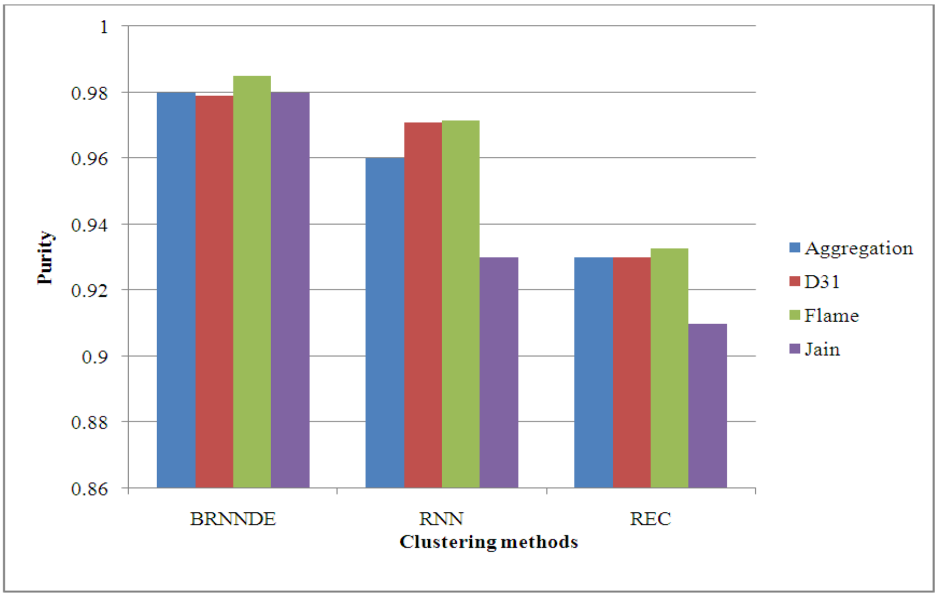

Figure3. Purity on Artificial Datasets with clustering methods

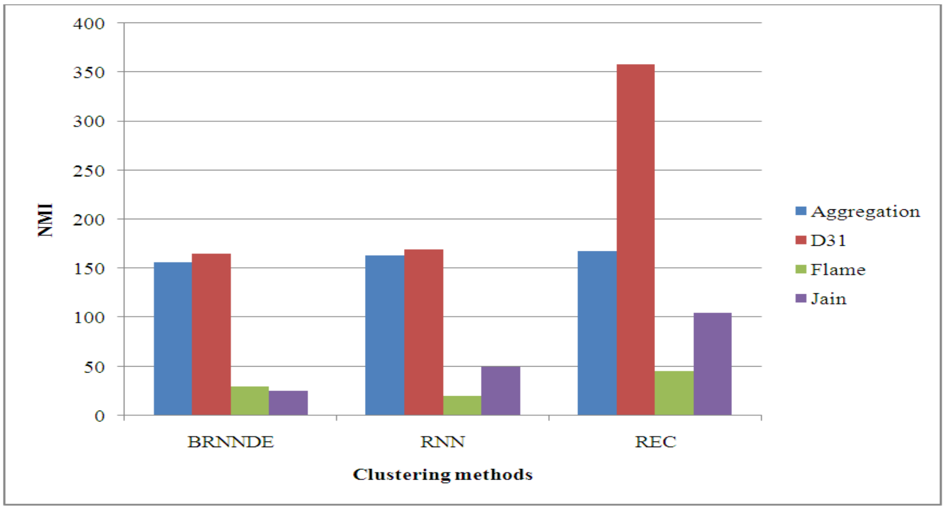

Figure4. NMI on Artificial Datasets with clustering methods

Figure 1-4 shows the results of various clustering methods such as DBSCAN-BRNNDE (BRNNDE), RNN-DBSCAN (RNN), and REC with respect to different metrics such as ARI, Purity, No. of cluster, and NMI on the artificial datasets. From these results, it concludes that the proposed BRNNDE clustering algorithm.

V. CONCLUSION AND FUTURE WORK

A novel density-based clustering algorithm, DBSCAN-BRNNDE, was presented using Bichromatic Reverse Nearest Neighbor based core observation and observation reachability definitions. One popular technique for improving the computational efficiency of density-based clustering is to combine it with BRNN clustering. Here feature space is partitioned into voronoi polygon which observations fall into same clusters. Density-based clustering is then performed on the voronoi such that observation is assigned to the cluster of their voronoi. A common voronoi polygon -based interpretation of proposed clustering algorithm, the existing clustering algorithms and DBSCAN was presented. Such an interpretation makes it easier to distinguish among the approaches with respect to several key components which include the graph definition, core observation identification, clustering by identifying connected components in some polygons, and extending clustering results. This last result is of significance given the reduced problem complexity of DBSCAN-BRNNDE with respect to DBSCAN (i.e., requires the use of a single parameter, k, as compared with the two parameters, eps and minpts). Future work will focus on the time characteristics of the objects to solve practical problems clustering the large-scale temporal data.

REFERENCES

[1]. Nagwani, N.K. and Bhansali, A., 2010. An object oriented email clustering model using weighted similarities between emails attributes. International Journal of research and reviews in Computer Science, 1(2), p.1.

[2]. Hinneburg, A., Keim, D.A.: An Efficient Approach to Clustering in Large Multimedia Databases with Noise. In: Proc. Int'l Conf. on Knowledge Discovery and Data Mining (KDD), New York, USA, pp. 58–65 (August 1998)

[3]. Ankerst, M., Breunig, M.M., Kriegel, H.-P., Sander, J.: OPTICS: Ordering Points To Identify the Clustering Structure. In: Proc. of Int'l Conf. on Management of Data, ACM SIGMOD, Philadelphia, Pennsylvania, USA, pp. 49–60 (June 1999).

[4]. Hinneburg, A., Gabriel, H.-H.: DENCLUE 2.0: Fast Clustering Based on Kernel Density Estimation. In: Berthold, M., Shawe-Taylor, J., Lavrač, N. (eds.) IDA 2007. LNCS, vol. 4723, pp. 70–80. Springer, Heidelberg (2007)

[5]. S. Vadapalli, S. R. Valluri, and K. Karlapalem, "A simple yet effective data clustering algorithm," in Sixth International Conference on Data Mining (ICDM'06), 2006, pp. 1108–1112.

[6]. C. Cassisi, A. Ferro, R. Giugno, G. Pigola, and A. Pulvirenti, "Enhancing density-based clustering: Parameter reduction and outlier detection," Inf. Syst., vol. 38, no. 3, pp. 317–330, 2013.

[7]. Y. Lv, T. Ma, M. Tang, J. Cao, Y. Tian, A. Al-Dhelaan, and M. Al- Rodhaan, "An efficient and scalable density-based clustering algorithm for datasets with complex structures," Neurocomput., vol.171, no. C, pp. 9–22, 2016.

[8]. Brecheisen, S., Kriegel, H.-P., Pfeifle, M.: Parallel Density-Based Clustering of Complex Objects. In: Ng, W.-K., Kitsuregawa, M., Li, J., Chang, K. (eds.) PAKDD 2006. LNCS (LNAI), vol. 3918, pp. 179–188. Springer, Heidelberg (2006)

[9]. Bohm, C., Noll, R., Plant, C., Wackersreuther, B.: Density-based Clustering using Graphics Processors. In: Proc. Conf. on Information and knowledge management (CIKM), Hong Kong, China, pp. 661–670 ,2009.

[10]. Louhichi, S., Gzara, M. and Ben-Abdallah, H., 2017. Unsupervised varied density based clustering algorithm using spline. Pattern Recognition Letters, 93, pp.48-57.

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

[12]. Su, X., Lan, Y., Wan, R. and Qin, Y., 2009, August. A fast incremental clustering algorithm. In International Symposium on Information Processing (pp. 175-178).

[13]. X. Chen, W. Liu, H. Qiu, and J.-H. Lai, "Apscan: A parameter free algorithm for clustering." Pattern Recognition Letters, vol. 32, no. 7, pp. 973–986, 2011.

[14]. Y. He, H. Tan, W. Luo, H. Mao, D. Ma, S. Feng, and J. Fan, "MRDBSCAN: An efficient parallel density-based clustering algorithm using mapreduce," IEEE 17th International Conference on Parallel and Distributed Systems, 2011, pp. 473–480.

[15]. B. Welton, E. Samanas, and B. P. Miller, "MR. SCAN: Extreme scale density-based clustering using a tree-based network of GPGPU nodes," Proceedings of the International Conference on High Performance

Computing, Networking, Storage and Analysis, ser. SC ’13. New York, NY, USA: ACM, 2013, pp. 84:1–84:11.

[16]. R. J. G. B. Campello, D. Moulavi, and J. Sander, Density-Based Clustering Based on Hierarchical Density Estimates. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 160–172.

[17]. Korn, F. and Muthukrishnan, S.: Influence Sets Based on Reverse Nearest Neighbor Queries. In: SIGMOD, pp. 201-212. Dallas, Texas, USA (2000)

[18]. Tran, Q.T., Taniar, D. and Safar, M.: Bichromatic Reverse Nearest-Neighbor Search in Mobile Systems. J. IEEE SYSTEMS. 4(2), pp. 230–242 (2010)

[19]. Kang, J.M., Mokbel, M.F., Shekhar, S., Xia, T. and Zhang, D.: Continuous Evaluation of Monochromatic and Bichromatic Reverse Nearest Neighbors. In: ICDE, pp. 806-815. Istanbul, Turkey (2007)

[20]. U. of Eastern Finland. Clustering datasets: Shape sets. [Online].Available: https://cs.joensuu.fi/sipu/datasets/

[21]. F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, “Scikit-learn: Machine learning in Python,” Journal of Machine Learning Research, vol. 12, pp. 2825–2830, 2011.

[22]. Strehl and J. Ghosh, “Cluster ensembles — a knowledge reuse framework for combining multiple partitions,” J. Mach. Learn. Res., vol. 3, pp. 583–617, Mar. 2003. [Online]. Available: http://dx.doi .org /10.1162/153244303321897735

[23]. T. M. Cover and J. A. Thomas, Elements of Information Theory. New York, NY, USA: Wiley-Interscience, 1991.

Mr.S.Premkumar. “DBSCAN-BRNNDE: A Density-based Clustering Algorithm using Bichromatic Reverse nearest Neighbor Density Estimates.” IOSR Journal of Engineering (IOSRJEN), vol. 08, no. 12, 2018, pp. 33-40.