International Journal of Scientific Research in Science, Engineering and Technology (www.ijsrset.com)
© 2020 IJSRSET | Volume 7 | Issue 1 | Print ISSN: 2395-1990 | Online ISSN : 2394-4099
DOI : https://doi.org/10.32628/IJSRSET207147

Empowering Density-based Micro-clusters In Dynamic Data Stream Clustering

Asha P. V.[1], Anju Sukumar[2]

[1]M. Tech Scholar, M Tech Scholar, Department of Computer Science and Engineering, Government Engineering College, Idukki, Kerala, India

[2]Assistant Professor, Department of Computer Science and Engineering, Government Engineering College, Idukki, Kerala, India

ABSTRACT

Data stream is a continuous sequence of data generated from various sources and continuously transferred from source to target. Streaming data needs to be processed without having access to all of the data. Some of the sources generating data streams are social networks, geospatial services, weather monitoring, e-commerce purchases, etc. Data stream mining is the process of acquiring knowledge structures from the continuously arriving data. Clustering is an unsupervised machine learning technique that can be used to extract knowledge patterns from the data stream. The mining of streaming data is challenging because the data is in huge amounts and arriving continuously. So the traditional algorithms are not suitable for mining data streams. Data stream mining requires fast processing algorithms using a single scan and a limited amount of memory. The micro clustering has a good role in this. In itself, density based micro clustering has its own unique place in data stream mining. This paper presents a survey on different data clustering algorithms, realizes and empowers the use of density-based micro clusters.

Keywords : Data Mining, Data Stream, Clustering.

I. INTRODUCTION

The clustering is a data mining technique that divides the input datasets into a convergence of subclasses or clusters. Fundamentally, the items that are present in one cluster are the same as one another and not at all like the items present in another cluster. Data modeling brings clustering in a historical prospect rooted in mathematics, statistics, and numerical examination. From an artificial intelligence perspective, clusters correspond to hidden patterns. The exploration for clusters is unsupervised learning, and the subsequent framework delivers a data concept. Clustering assumes a regarded job in many data mining applications. Some of the applications are exploration scientific data, data recovery and content mining, spatial database applications, and many others. Data stream mining is not quite the same as conventional data mining. It carries additional issues to the conventional technique. A data stream is possibly an unbounded sequence of data. In dynamic environments, the properties of this data can change over time in unexpected ways. The performance of traditional classifiers and predictive models can debase as the stream progresses because the characteristics of the objective articles change. This change might be progressive, known as concept-drift, sudden as concept-shift, or in the form of concept-evolution when new classes turn out in the stream. Recognizing this change is a challenge, as is

responding and modifying to the change. This challenge is related with many issues like the shortage-of-labels, cost of recently arriving data, and time taken to label. Unsupervised learning techniques, such as clustering, can potentially be used to diminish this labeling issue and also as a change detection system at the side of customary classifiers and predictive models.

Clustering is widely used in many applications such as statistical surveying, pattern recognition, data analysis, and picture handling. It can also help marketers identify perceived groups in their customer base and they can characterize their customer groups dependent on the buying patterns. In the field of biology, it can be used to derive plant and animal scientific classifications, categorize genes with similar functionalities and gain insight into structures innate to populations. It also helps in cataloging of areas of similar land use in an earth observation database. It can also be used in the identification of groups of houses in a city according to house type, value, and geographic location. Clustering helps to classify documents on the web for information discovery. It is also used in outlier detection applications such as detection of credit card fraud. Cluster analysis fills in as a device to accomplish understanding into the distribution of data to observe the characteristics of each cluster.

Clustering is a functioning exploration subject in different fields. This study is centered around clustering in data mining. Clustering becomes convoluted by enormous datasets with endless attributes of different sorts. This enforces exclusive computational prerequesites on pertinent clustering algorithms. Various algorithms have as of late built up that meet these prerequesites and were effectively applied to real-life data mining problems. They are the subject of the survey.

A. Data Stream Clustering

Most likely the clustering algorithms generate clusters over the entire data set. Every one of these algorithms treat clustering as a single-pass grouping method. Such clustering strategies are useful for some applications, however, in the case of data streams, it is necessary to define the clustering problem carefully because the data stream is an infinite process in which data is evolved along with time. The clusters are also varying concerning the time when they are determined and the time over which they are estimated. For instance, a specific application may have the requirement of the user to look at clustering occurring concerning a time-frame, these clusters are attended as a special case. Therefore, the data stream algorithm must provide an interface for user-defined periods to generate relevant clusters. Various surveys have been conducted on mining data streams. A comparison is made between different categories of data clustering. The goal is to find the best performing data stream clustering algorithm.

Clustering in a data stream is entirely different from traditional clustering. When dealing with a continuous sequence of information, it may be allowed to examine the data once. In order to prevent obstructions and potential loss of data, Clustering needs to be performed quickly. A stream can be potentially infinite but only a limited amount of memory is available, necessitating the summarization of identified clusters in a meaningful way. According to the nature of an evolving stream, clusters can drift, new clusters can appear, or clusters can disappear and reappear cyclically. Therefore, it is difficult to know a priori how many clusters are present in the stream. Although traditional partitioning clustering techniques, such as k-means and its variants have been successfully applied to data streaming, they have the drawback of requiring k to be specified a priori. Density-based clustering, a form of hierarchical clustering, defeats this limitation.

B. Density-Based Clustering

In many of the real-life applications, some of the data set may contain large amounts of noise and outliers. Therefore clusters are not really of spherical shape. The noise and outliers are adequately handled by the density-based clustering methods. It is not necessary to indicate the number of clusters as initial assumptions. In many applications, the main issue is to do a cluster analysis of evolving data streams.
The density-based method is a noteworthy class in clustering data streams, which can discover arbitrary shape clusters and to detect noise. Furthermore, the number of clusters is not needed in advance. The traditional density-based clustering is not applicable, due to data stream characteristics. In recent times, enough number of density-based clustering algorithms are extended for data streams. Using density-based methods in the clustering process is the main idea of these algorithms and at the same time overwhelming the constraints, which are put out by the data stream's nature.

Density-based clustering defines clusters as high-density areas of the feature space separated by areas of low density. It can identify arbitrarily shaped clusters, is robust to outliers and, necessarily, does not require the number of clusters to be known a priori. In our proposed algorithm, dense areas are described using micro-clusters.

C. Density-Based Micro-Clustering

The outstanding density-based clustering algorithms based on micro-clusters are analyzed here. Micro-clusters are n-dimensional spheres with center c and radius r. Micro-clusters have a maximum radius where $r \leq \varepsilon$. Generally a data point is assigned to a micro-cluster if the point falls within its radius. The set of micro-clusters that are connected form the macro-cluster. The number of micro-clusters is generally more than the actual clusters but significantly fewer micro-clusters than there are data points. This serves a dual purpose both as the clustering mechanism and as a summarization technique because many local data points can be represented by a single micro-cluster. Clusters identified by the algorithm are summarized by their constituent micro clusters and these summaries are stored offline for evaluation by the user. This has two advantages: The first is information about clusters that can be stored in a fraction of the space. The second is that it is easier to evaluate representative micro-clusters than the entire set of individual data points assigned to a cluster.

A density-based approach to stream clustering can address the problem of a shifting number of non-stationary clusters and provides a method to summarize these clusters, and we propose a sampling method to address the speed requirement of data stream clustering. A point's similarity with a cluster is evaluated using a sample taken from the cluster. The stochastic sampling method replaces the traditional, exhaustive search for each point's appropriate micro-cluster, and subsequently the nearest neighbor of each micro-cluster. Rough clusters are created incrementally in a single pass of the data. The first point seeds the first clu ster, subsequent points are assigned to an existing cluster or, if too dissimilar, seed a new cluster. The micro-clusters are created only after every point has been assigned to its respective cluster. Each point is converted to a micro-cluster and these micro-clusters attempt to merge with others in the same cluster only. Attempting to merge at this stage reduces the number of failed merging attempts. Keep in mind that the merging operation is expensive.

After this single-pass of the data, the clusters now discovered are often rough and too many in number. These clusters are refined using a sorting method. Sorting ants are assigned to each cluster and they attempt to refine the initial clusters by probabilistically picking micro-clusters and dropping them in more suitable clusters.

II. RELATED WORKS

H. Azzag et al. [1] presented a new clustering algorithm for unsupervised learning. It is based on the self-assembling behavior of real ants. The ants progressively become attached to an existing support and then successively to other attached ants. Similarly, a tree will be built by the artificial ants that they have defined. Each ant represents one data. The way ants move and build this tree is completely based on the similarity between the data. The results obtained are compared with the k-means algorithm and by AntClass on numerical databases (either artificial, real). It shows that AntTree significantly improves the clustering process.

Charu C. Aggarwal et al.[2] presented a fundamentally different philosophy for data stream clustering which is guided by application-centered requirements. The idea is to divide the clustering process into an online component and an offline component. The online component periodically stores detailed summary statistics. The offline component is utilized by the analyst, who can use a wide variety of inputs to provide a quick understanding of the broad clusters in the data stream. The problems of efficient choice, storage, and use of this statistical data for a fast data stream is quite tricky. For this purpose, they used the concepts of a pyramidal time frame in conjunction with a micro clustering approach. Their performance experiments over several real and synthetic data sets illustrate the effectiveness, efficiency, and insights provided by their approach.

Thomas A. Runkler et al. [3] simplified the original ant system algorithm and provided a generalized ant colony optimization algorithm that can be used to solve a wide variety of discrete optimization problems. It is shown hard and fuzzy c-means can be optimized using particular extensions of this simplified ant optimization algorithm. Experiments with artificial and real datasets show that ant clustering produces better results than alternating techniques.

Martin Ester et al. [4] proposed a new approach for discovering clusters in an evolving data stream. The core-micro-cluster is introduced to outline the clusters with discretionary shape, while the potential core-micro-cluster and outlier micro-cluster structures are proposed to maintain and distinguish the potential clusters and outliers. A different pruning approach is designed based on these concepts, which assures the precision of the weights of the micro-clusters with limited memory. The experimental performance evaluation over several real and synthetic data sets demonstrates the effectiveness and efficiency of DenStream in discovering clusters of arbitrary shape in data streams.

Luning Xia Jiwu Jing et al. [5] introduced the concept of clustering ensemble to avoid the difficulty of selecting a single appropriate threshold. Operating DBSCAN many times with distinct thresholds picked up from a pre-constructed interval, the final partition can be figured out via a consensus function. Experimental results show that this method can go above DBSCAN both in the validity and stability, and bypass the inefficiency caused by any improper thresholds.

Philipp Kranen et al. [6] proposed a work in which they proposed a parameter-free algorithm that automatically adapts to the speed of the data stream. Instead of storing all incoming objects, a cluster feature tuple CF = (n, LS, SS) of the number n of represented objects, their linear sum LS, and their squared sum SS is maintained. Any cluster feature (CF) then represents a micro-cluster. They propose maintaining cluster features (CFs) by extending index structures from the R-tree family. Such hierarchical indexing structures provide the techniques for accurately locating the right place to insert any object from the stream into a micro-cluster. The idea is to build a hierarchy of micro-clusters at different levels

of granularity. If this micro-cluster is related enough, it is amended incrementally by this object's values. Otherwise, a new micro-cluster may be formed.

Rashmi Dutta Baruah and Plamen Angelov [7] developed a new online evolving clustering approach for streaming data is proposed. This approach uses cluster weight and distance before making new clusters. The dynamics of the data stream is defined by the cluster weight. Which is described in both data and time-space in such a way that it decays exponentially with time. Concepts from computational geometry is also applied to determine the neighborhood information while forming clusters. The real outliers can be effectively identified by making a distinction between core and noncore clusters. The approach not only adds rules but also removes them and thus maintains a lower complexity of models. This not only saves memory space but also reduces the estimation time. Furthermore, it enhances the speed by detecting and removing noncore clusters at the appropriate time.

Nesrine Masmoudi et al. [8] presented in this paper a new bio-inspired algorithm that dynamically creates groups of data. This algorithm is established on the concept of artificial ants that move together in a complex way with simple localization rules. Each ant represents one datum in the algorithm. The moves of ants aim at creating homogeneous groups of data that evolve together in a graph environment. He also suggests an extension of this algorithm to treat data streaming. That extended algorithm has been tested on real-world data. Their algorithms yielded competitive results as compared to K-means and Ascending Hierarchical Clustering (AHC), two well-known methods. Their method is based on ants system is well known to be adaptive and efficient to dynamical situations.

Shengxiang Yang et al. [9] presented a bio-inspired approach to clustering non-stationary data streams. The proposed algorithm, Ant-Colony Stream Clustering (ACSC), is based on the concept of artificial ants. That identifies clusters as nests of micro-clusters in dense areas of the data. Micro-clusters are defined as N-dimensional spheres with a maximum radius ε. In ACSC the ε-neighborhood, crucial in density clustering is adaptive and doesn't require expert, data-dependent tuning. The sliding window model is used in this algorithm and summary statistics for each window are stored offline. From the experimental results over real and synthetic data sets, it is found that the clustering quality of ACSC is promising than leading stream-clustering algorithms. Moreover, less computation is required with fewer parameters.

Conor Fahy et al. [10] presented an online, bio-inspired approach to clustering dynamic data streams. Their proposed Ant Colony Stream Clustering (ACSC) algorithm is a density-based grouping approach. It identifies clusters as high-density areas of the feature space separated by low-density areas. ACSC identifies clusters as groups of micro-clusters. The stream of data is read by a tumbling window. Initial clusters are incrementally formed during a single pass. Several points can be moved in a single operation. This speeds up the algorithm with just a minor expense to execution .The rough clusters are then refined using a method inspired by the observed sorting behavior of ants. Ants pick-up and drop items depends on the similarity with the surrounding items. Artificial ants sort clusters by picking and dropping micro-clusters based on local density and local similarity. Micro clusters summary statistics are stored offline and are used for creating final clusters. It is clear from the experimental results that the clustering quality of ACSC is versatile, robust to noise and higher than leading ant clustering algorithms. It also requires fewer parameters and less computational time.

Table 1. Comparison of Clustering Algorithms

Sl.No.	Title	Year	Method used	Advantages	Limitations
1	AntTree: a New Model for Clustering with Artificial Ants	2003	Tree-structured organization of the data.	Better than K-means and AntClass algorithms(lower clustering errors)	Ants cannot be moved to a much similar position if exists.
2	A Framework for Clustering Evolving Data Streams	2003	CluStream algorithm is used.	CluStream can achieve higher accuracy than STREAM	Use of k-means makes the algorithm inconsistent.
3	Ant Colony Optimization of Clustering Models	2005	A fuzzification of the ACO algorithm is used.	Clustering produces better results than alternating optimization techniques.	ACO seems to be not suited for soft clustering.
4	Density-Based Clustering over an Evolving Data Stream with Noise	2006	The density-based clustering method is used.	High clustering quality compared to DBSCAN.	It does not release any memory space by either deleting a micro-cluster or by merging two old micro-clusters.
5	An Ensemble Density-based Clustering Method	2007	Basic DBSCAN with ensembling is used.	The stability of EDBSCAN is far better than that of DBSCAN	The speed of EDBSCAN is slower than that of DBSCAN.
6	The ClusTree: indexing micro-clusters for anytime stream mining	2010	ClusTree algorithm is used.	Any time stream clustering is possible.	Computational complexity is a disadvantage.
7	DEC: Dynamically Evolving Clustering and Its Application to Structure Identification of Evolving Fuzzy Models	2014	The clustering method uses cluster weight and distance measures.	Enhances the speed by detecting and removing noncore clusters at the appropriate time.	The selection of optimal radius is an issue
8	How to Use Ants for Data Stream Clustering	2015	Based on the concept of artificial ants	Better results compared to K-means and Ascending Hierarchical Clustering (AHC)	Hierarchical clustering is not used.
9	Dynamic Stream Clustering Using Ants	2016	Bio-inspired approach, based on the concept of artificial ants	The clustering quality is high.	The sliding window is used instead of a tumbling window.
10	Ant Colony Stream	2019	Density-based	The clustering	The density

	Clustering: A Fast Density Clustering Algorithm for Dynamic Data Streams		clustering algorithm. Hierarchical clustering is used.	quality of ACSC is scalable. Robust to noise. Less computational time.	difference among clusters is not normalized.

III.CONCLUSION

In this paper, several algorithms proposed for clustering have been discussed. A review has been made on various clustering algorithms that are using data streams. All are discussed along with their advantages and disadvantages. This survey can be helpful for the understanding of several clustering algorithms for choosing the appropriate algorithm. The type of algorithm that is to be chosen depends upon the type of clusters that are needed to be found, type of data set and many attributes. In this survey, a group of clustering methods have been analyzed. It has been found that ACSC was shown to outperform other clustering algorithms in the literature. Further betterment can be achieved in ACSC by introducing techniques like density normalization of the previously formed micro clusters.

IV. REFERENCES

[1]. H. Azzag, N. Monmarche, M. Slimane, and G. Venturini, "AntTree: A new model for clustering with artificial ants," in Proc. IEEE Conf. Evol. Comput., vol. 4. Canberra, ACT, Australia, 2003, pp. 2642–2647

[2]. Charu C. Aggarwal, T. J. Watson Resch. Ctr.Jiawei Han, Jianyong Wang, UIUC, Philip S. Yu, T. J. Watson Resch. Ctr., "A Framework for Clustering Evolving Data Streams," Proceedings of the 29th VLDB Conference, Berlin, Germany, 2003.

[3]. Thomas A. Runkler Siemens AG Corporate Technology, Information, and Communications, 81730 Munich, Germany, " Ant Colony Optimization of Clustering Models," in INTERNATIONAL JOURNAL OF INTELLIGENT SYSTEMS, VOL. 20, 1233–1251, 2005.

[4]. F. Cao, M. Ester, W. Qian, and A. Zhou, "Density-based clustering over an evolving data stream with noise," in Proc. SIAM Int. Conf. D ata Min., vol. 6, 2006, pp. 328–339.

[5]. Luning Xia Jiwu Jing, "An Ensemble Density-based Clustering Method," Information Security State Key Laboratory, Graduate University of Chinese Academy of Science, Beijing 100049, P. R. China,2007.

[6]. P. Kranen, I. Assent, C. Baldauf, and T. Seidl, "The ClusTree: Indexing micro-clusters for any time stream mining," Knowl. Inf. Syst., vol. 29, no. 2, pp. 249–272, 2011.

[7]. R. D. Baruah and P. Angelov, "DEC: Dynamically evolving clustering and its application to structure identification of evolving fuzzy models," IEEE Trans. Cybern., vol. 44, no. 9, pp. 1619–1631, Sep. 2014.

[8]. Nesrine Masmoudi, Hanane Azzag, Mustapha Lebbah, Cyrille Bertelle, Maher Ben Jemaa, "How to Use Ants for Data Stream Clustering," in Proc. IEEE Conf. Evol. Comput., vol. 4. Canberra, ACT, Australia2015.

[9]. Conor Fahy and Shengxiang Yang, "Dynamic Stream Clustering Using Ants," Published in UKCI 2016.

[10]. Conor Fahy, Shengxiang Yang, Senior Member, IEEE, and Mario Gongora, "Ant Colony Stream Clustering: A Fast Density Clustering Algorithm for Dynamic Data Streams," in IEEE TRANSACTIONS ON CYBERNETICS, VOL. 49, NO. 6, JUNE 2019.

Cite this article as :

Asha P. V., Anju M. Sukumar, "Empowering Density-based Micro-clusters In Dynamic Data Stream Clustering", International Journal of Scientific Research in Science, Engineering and Technology (IJSRSET), Online ISSN : 2394-4099, Print ISSN : 2395-1990, Volume 7 Issue 1, pp. 259-265, January-February 2020. Available at doi : https://doi.org/10.32628/IJSRSET207147

Journal URL : http://ijsrset.com/IJSRSET207147