High Speed Clustering Scheme for High Dimensional Data Streams

Sudeesh. S[1] M. Suresh[2]
[1]Research Student [2]Associate Professor
[1,2]Department of Computer Science & Engineering
[1,2]Selvam College of Technology, Namakkal, Tamilnadu, India

Abstract— This paper presents a novel high speed clustering scheme for high dimensional data streams. Data stream clustering has gained importance in different applications, for example, in network monitoring, intrusion detection, and real-time sensing are few of those. High dimensional stream data is inherently more complex when used for clustering because the evolving nature of the stream data and high dimensionality make it non-trivial. In order to tackle this problem, projected subspace within the high dimensions and limited window sized data per unit of time are used for clustering purpose. We propose a High Speed and Dimensions data stream clustering scheme (HSDStream) which employs exponential moving averages to reduce the size of the memory and speed up the processing of projected subspace data stream. The proposed algorithm has been tested against HDDStream for cluster purity, memory usage, and the cluster sensitivity. Experimental results have been obtained for corrected KDD intrusion detection dataset. These results show that HSDStream outperforms the HDDStream in performance metrics, especially the memory usage and the processing speed.

Key words: Clustering. Data Stream, High Dimensionality

I. INTRODUCTION

THE exponential growth in data mining and clustering is an apparent result of network applications that are becoming part of our daily life. In today's applications, whether they are related to academic, research, finance, business, or military, the evolving data streams are ubiquitous. Data sources are monotonically increasing from past few decades. Additionally, the technological developments in data sensing systems (sensor networks) have resulted in a real-time data with large number of attributes. The large volume of the data together with its high dimensionality has motivated the research in the area of high dimensional data mining and exploration. Data stream is a form of data that continuously evolves reflecting the real-time variation in volume, dimensionality, and correlation. In recent years, a large amount of streaming data, such as network flows, wireless sensor networks data and the multimedia streams have been generated. The irrelevant attributes interfere with the efforts to find targeted clusters. This problem is become more intensive in streaming data, because it requires a single scan of the data to find the useful attributes for describing a potential cluster for the current object. Moreover, streams are impulsive and the discovered clusters might also evolve over time. High dimensional streaming data clustering is more challenging than the high density or high dimensional data. Generally, there are two types of stream clustering algorithms: full dimensional and projected or preferred dimension streaming algorithms. Clustering applications in various domains often have very high-dimensional data. Storage and time limits are critical for clustering algorithms to perform a fast single- pass over that stream data.

II. RELATED WORK

In the last few years many research works have been done on high dimensional data clustering and evolving data streams clustering. There are extensive research works on clustering algorithms for static datasets where some of them have been further extended for evolving data streams. The clusters are formed based on a Euclidean distance function like k-means algorithm. K-mean clustering splits the n d-dimensional points into k cluster (k < n). One of the well-known extensions of k-means on data streams is presented by Aggarwal et al. They proposed an algorithm called CluStream based on k-means for clustering evolving data streams. CluStream introduces an online-offline method for clustering data streams. CluStream clustering idea has been adopted for the majority of data stream clustering algorithms. Aggarwal et al. extended their work in HPStream, which introduces the projected clustering to data streams. In projected clustering high dimensional stream data has been partitioned based on preferred dimensions instead of full dimensional space. Cao et al. use the density-based clustering without projected dimensions in DenStream algorithm. For streaming data, although a considerable research has tackled the full-space clustering, relatively limited work has been dealt with subspace clustering. These few researches include HPStream, [11] HDDStream, and SubCMM. A more comprehensive review and classifications are given in survey. In authors proposed a density-based projected clustering scheme for high dimensional data streams called HDDStream. HDDStream works in three phases; an initial phase in which initial set of core micro-clusters has been formed, then online core and outlier clusters' maintenance with projected clustering, and finally, an on-demand offline clustering phase. Compared with HPStream which requires the fixed number of clusters, the number of clusters in HDDStream is variably adjusted over time, and the clusters can be of arbitrary shape. SubCMM suggests a different way for evaluating stream subspace clustering algorithms by making use of available offline subspace clustering algorithms with the streaming environment to handle the errors caused by emerging, moving, or splitting subspace clusters. A recent, similarity-based Data StreamClassifier (SimC) introduces an insertion/removal policy that adapts evolving data tendency and maintains a representative, small set of clusters. It uses instance based learning techniques to form adaptive clustering algorithm. In clustering method based on a multi-agent system that uses a decentralized bottom-up self-organizing strategy to group similar data points has been presented. It uses bio-inspired flocking model to eliminate the need of offline clustering. A clustering algorithm for stream data with uncertain attributes has been presented in. This scheme works only for low dimensional streaming data. Liu developed HSWStream algorithm. It is a data stream clustering algorithm based on exponential histogram over sliding windows with projected dimensions. Another density-

based algorithm D-Stream maps each input data into a grid, computes the density of each grid, and forms the clusters using these grids. In authors proposed a scalable algorithm to trace clusters in a high-dimensional data stream. The proposed scheme transforms the problem of multi-dimensional clustering into that of one-dimensional clustering along with a frequent itemset mining technique. This scheme achieves the scalability on the number of dimensions while sacrificing the accuracy of identified clusters.

III. PROBLEM FORMULATION

In general, data stream is modeled as an infinite series of points {p1, p2, ..., pi, ...} arriving at discrete time {t1, t2, ...ti, ...}. Each point pi is a vector of dimension d such that pi = {pi,1, pi,2, ..., pi,d}. An important characteristic of data streams is that we cannot store all data points. A usual way to overcome this problem is to summarize the data through an appropriate summary structure, often called micro-cluster. A microcluster summarizes the time and dimensionality limited stream data in the form of a tuple. When aging is also under consideration, the temporal extension of micro clusters.

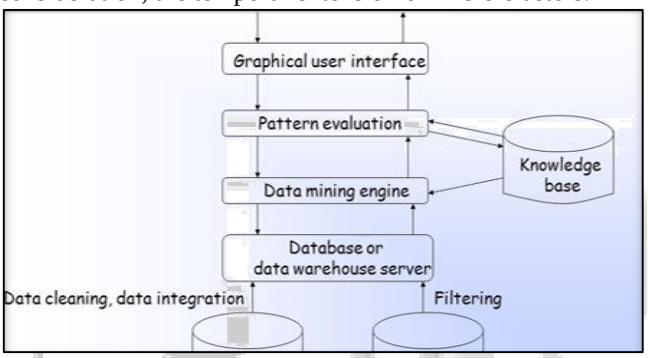

Fig. 1: System Architecture

IV. THE HSDSTREAM ALGORITHM

HSDStream algorithm can be divided into three parts: 1) Initialization to produce a set of representative core Microcluster (core-mc) from an initial chunk of data points, 2) online maintenance of core-mc and outlier micro-cluster (outlier-mc), and, 3) offline generating the final clusters, on demand by the user.

A. Initialization

In order to get initial set of micro-clusters from a fixed size of data points, we apply density-based projected clustering algorithm, a variant of PreDeCon algorithm, which is designed to work for fixed size of data of high dimensionality. Let D be a set of initial chunk of d-dimensional data points ($D \subseteq Rd$). For each point $p \in D$, we find a set of ϱ−neighbors $N_{\varrho}(p)$. In addition to this, we find the neighbors of p with projected distance equal to or less than the ϱ, namely, $N(p)_{\varrho}$ (p).

B. Real-time Maintenance of Micro-clusters

In order to find out the clusters in an evolving real-time data stream, we maintain two groups of micro-clusters, namely, core-mc and outlier-mc in real-time. All the micro-clusters are maintained in a separate memory space. A new point might be assigned to core-mc, outlier-mc, or it may start new outlier-mc depends upon various factor.

Sequential process of merging a new point p is described below:

- When a new point arrives, it first becomes the candidate of core-mc. The projected dimensionality of each core-mc has been evaluated before and after adding this point p (algorithm 2, line 4). After that, projected distance of p is calculated with those core-mc which still satisfy the projected dimensionality constraint, i.e., after the addition of point p (algorithm2, line 6).
- When a new point becomes a candidate for an outliermc, the projected distance of p with each outlier-mc has been evaluated (algorithm 3, line 4). The closest distant outlier-mc is chosen in line 6. The point p becomes the member of that outlier-mc if the projected radius is less than or equal to the radius threshold (ϱ) (algorithm3, line 9).
- If point p cannot be added in core-mc or outlier-mc (algorithm 3, line 14) then a new outlier-mc is created with this point being the first element. It may become the seed of future core-mc.

C. Cluster Quality Evaluation

Traditional full dimensional clustering algorithms, for example, used the sum of square distances (SSQ) to evaluate the clustering quality. However, SSQ is not a good measure in evaluating projected clustering [9] because it is a full dimensional measure, and full dimensional measures are not very useful for measuring the quality of a projected clustering algorithm. So, as in [9] and [11], we evaluate the clustering quality by the average purity of clusters, which examines the purity of the clusters with respect to the true cluster (class) labels. During the time interval from 250 to 365 we encounter with several attacks (back, ipsweep, nmap, and neptune) along with correlated normal data so that we can see cluster purity is equal to 1 for this time interval. Satan attacks the network from 453 to 455 time units, followed by smurf attack which continues till the end of simulations at 495 time units. It can be observed that HDDStream has the same purity graph pattern as HSDStream but with considerably low magnitude.

D. Clusters Generation: Offline

The real-time maintained micro-clusters capture the density area and the projected dimensionality of data streams. However, in order to get meaningful clusters, we need to apply some clustering algorithm to get the final result. When a clustering request arrives, a variant of PreDeCon algorithm is applied on the set of real-time maintained coremc(s) to get the final result of clustering.

E. Memory Usage

We measure the memory usage as a number of microclusters in HDDStream and HDSStream. During the period of highly correlated normal data or the network attack, there is only one core-mc containing all the correlated points and no outlier cluster exists. When we compare these figures with different window sizes, we can see that there is a gradual increase of number of clusters with increasing number of window size. HSDStream outperforms the HDDStreamin terms of memory usage for all window sizes, which is due to our reduced memory sized tuple and high density micro-clusters.

V. DISCUSSION

In this section we highlight issues and challenges in the development of high dimensional data stream clustering in Internet traffic monitoring. We maintain the density with ϱ−neighborhood and minimum number of point's μ in a core-mc. When an identical burst of data (in case of attack on network) arrives, outlier-mc(s) are diminished and only one core mc remains there. In this case, an important entity of core-mc formation i.e., projected dimensionality cannot work because, now PDIM = d and it no longer satisfies the condition PDIM ≤ π. In order to overcome this problem we introduce another condition ORed with the condition PDIM ≤ π to maintain one core-mc containing exactly similar data. The new condition is W(t)/N > 90% , i.e., if the data points window contains more than 90% points, then no need to check PDIM because themajority of identical data points indicates some abnormal activity on the network being monitored.

VI. CONCLUSION

This paper presents a clustering algorithm for high dimensional high density streaming data. We propose a new structure of micro-cluster's tuples. This structure uses exponential weighted averages to reduce the memory usage and decrease the computational complexity. We have compared our scheme with HDDStream with KDD network intrusion detection dataset. The results show that HSDStream give significant improvement over HDDStream in terms of cluster purity, memory usage, and the processing time.

REFERENCES

[1] A. Forestiero, C. Pizzuti, and G. Spezzano, "A single pass algorithm for clustering evolving data streams based on swarm intelligence," Data Mining and Knowledge Discovery, vol. 26, no. 1, pp. 1–26, Jan. 2013. [Online]. Available: http://link.springer.com/10.1007/s10618-011-0242-x.

[2] K. Sim, V. Gopalkrishnan, A. Zimek, and G. Cong, "A survey on enhanced subspace clustering," Data Mining and Knowledge Discovery, vol. 26, no. 2, pp. 332–397, Mar. 2013. [Online]. Available: http://link.springer.com/10.1007/s10618-012-0258-x.

[3] C. C. Aggarwal, "A segment-based framework for modelling and mining data streams," Knowledge and Information Systems, vol. 30, no. 1, pp. 1–29, Jan. 2012. [Online]. Available: http://link.springer.com/10.1007/s10115-010-0366-0.

[4] A. Amini, T. Y. Wah, and H. Saboohi, "On density-based data streams clustering algorithms: A survey," Journal of Computer Science and Technology, vol. 29, no. 1,pp.116141,2014.[Online].Available: http://link.springer.com/article/10.1007/s11390014-1416-y.

[5] A. K. Jain, "Data clustering: 50 years beyond k-means," Pattern Recognition Letters, vol. 31, no. 8, pp. 651–666, Jun. 2010. [Online]. Available: http://linkinghub.elsevier.com/retrieve/pii/S0167865509002323.

[6] H.-P. Kriegel, P. Krger, and A. Zimek, "Clustering high dimensional data: A survey on subspace clustering, pattern-based clustering, and correlation clustering," ACM Trans. Knowl. Discov. Data, vol. 3, no. 1, pp. 1:1–1:58, Mar. 2009. [Online]. Available: http://doi.acm.org/10.1145/1497577.1497578.

[7] J. MacQueen, "Some methods for classification and analysis of multivariate observations." The Regents of the University of California, 1967. [Online]. Available: http://projecteuclid.org/euclid.bsmsp/1200512992.

[8] C. C. Aggarwal, J. Han, J. Wang, and P. S. Yu, "A framework for clustering evolving data streams," in Proceedings of the 29th international conference on Very large data bases-Volume 29. VLDB Endowment, 2003, pp. 81–92. [Online]. Available: http://dl.acm.org/citation.cfm?id=1315460.

[9] "A framework for projected clustering of high dimensional data streams," in Proceedings of the Thirtieth international conference on Very large data bases-Volume 30.VLDB Endowment, 2004, pp. 852–863. [Online]. Available: http://dl.acm.org/citation.cfm?id=1316763.

[10] F. Cao, M. Ester, W. Qian, and A. Zhou, "Density-based clustering over an evolving data stream with noise." In SDM, vol. 6. SIAM, 2006, pp. 326–337. [Online]. Available: http://epubs.siam.org/doi/abs/10.1137/1.9781611972764.29.

[11] I. Ntoutsi, A. Zimek, T. Palpanas, P. Krger, and H.-P. Kriegel, "Density-based projected clustering over high dimensional data streams." in SDM. SIAM, 2012, pp. 987–998. [Online]. Available: http://epubs.siam.org/doi/abs/10.1137/1.9781611972825.85.

[12] M. Hassani, Y. Kim, S. Choi, and T. Seidl, "Subspace clustering of data streams: new algorithms and effective evaluation measures," Journal of Intelligent Information Systems, Jun. 2014. [Online].Available: http://link.springer.com/10.1007/s10844-014-0319-2.

[13] H.-L. Nguyen, Y.-K. Woon, and W.-K. Ng, "A survey on data stream clustering and classification," Knowledge and Information Systems, Dec. 2014. [Online]. Available: http://link.springer.com/10.1007/s10115-014-0808-1.

[14] D. Mena-Torres and J. S. Aguilar-Ruiz, "A similarity-based approach for data stream classification," Expert Systems with Appli-cations, vol. 41, no. 9, pp.4224–4234, Jul. 2014. [Online]. Available: http://linkinghub.elsevier.com/retrieve/pii/S0957417413010300.

