Neha Gupta, Indrjeet Rajput / International Journal of Engineering Research and Applications (IJERA) ISSN: 2248-9622 www.ijera.com
Vol. 3, Issue 1, January -February 2013, pp.1113-1118

Stream Data Mining: A Survey

Neha Gupta*[1], Indrjeet Rajput*[2]
*Department of Computer Engineering,Gujarat Technological University, Gujarat

ABSTRACT

A data stream is a massive, continuous and rapid sequence of data elements. Mining data streams raises new problems for the data mining community about how to mine continuous high-speed data items that you can only have one look at. Due to this reason, traditional data mining approach is replaced by systems of some special characteristics, such as continuous arrival in multiple, rapid, time-varying, possibly unpredictable and unbounded. Analyzing data streams help in applications like scientific applications, business and astronomy etc.

In this paper, we present the overview of the growing field of Data Streams. We cover the theoretical basis needed for analyzing the streams of data. We discuss the various techniques used for mining data streams. The focus of this paper is to study the problems involved in mining data streams. Finally, suggested we conclude with a brief discussion of the big open problems and some promising research directions in the future in the area.

Keywords- **Data Streams, Association Mining, Classification Mining, Clustering Mining.**

I. INTRODUCTION

As Data mining is considered as a process of discovering useful patterns beneath the data, also uses machine learning algorithms. There have been techniques which used computer programs to automatically extract models representing patterns from data and then check those models. Traditional data mining techniques cannot be applied to data streams. Due to most required multiple scans of data to extract information, for stream data it was unrealistic. Information systems have been more complex, even amount of data being processed have increased and dynamic also, because of common updates. This streaming information has the following characteristics.

• The data arrives continuously from data streams.
• No assumptions on data stream ordering can be made.
•The length of the data stream is unbounded.

Efficiently and effectively capturing knowledge from data streams has become very critical; which include network traffic monitoring, web click-stream There is need of employment of semi-automated interactive techniques for the extraction of hidden knowledge and information in the real-time. Systems, models and techniques have been proposed and developed over the past few years to discuss these challenges [1, 3].

The remaining parts of this paper are organized as follows: In Section 2, we briefly discuss the necessary background needed for data stream analysis. Then, Section 3 and 4 describes techniques and systems used for mining data streams open and addressed research issues in this growing field are discussed in Section 5. These research issues should be addressed in order to realize robust systems that are capable of fulfilling the needs of data stream mining applications. Finally, section 6 summarizes the paper and discusses interesting directions for future research.

II. ESSENTIAL METHODOLOGIES OF DATA STREAMS

From the well-established statistical and computational approaches the problems of mining data streams can be solved using the methodologies which uses

• Examining a subset of the whole data set or transforms the data to reduce the size.
• Algorithms for efficient utilization of time and space, detailed discussion follow in section 3.

The First methodology refers to summarization of the whole data set or selection of a subset of the incoming stream to be analyzed. The techniques used are Sampling, load shedding, sketching, synopsis data structures and aggregation. These techniques are briefly discussed here.

2.1 Sampling

Sampling is one of the oldest statistical techniques, used for a long time which makes a probabilistic choice of data item to be processed. Boundaries of error rate of the computation are given as a function of the sampling rate. Sampling is considered in the process of selecting incoming stream elements to be analyzed .Some computing approximate frequency counts over data streams previously performed by sampling techniques. Very Fast Machine learning techniques [2] have used Hoeffding bound to measure the size of the sample. Even sampling approaches had been studied for clustering data streams, classification techniques, and the sliding window model. [7][8].

Problems related to the sampling technique while analyzing the data stream are

• Unknown size of data set

• Sampling may not be the correct choice in applications which check for anomalies in surveillance analysis.
• It does not address the problem of fluctuating data rates.
Relations among data rate, sampling rate and error bounds are to be generated.

2.2 Load shedding
Load shedding refers to the process of dropping a fraction of data streams during periods of overload. Load shedding is used in querying data streams for optimization. It is desirable to shed load to minimize the drop in accuracy. Load shedding also has two steps. Firstly, choose target sampling rates for each query. In the second step, place the load shedders to realize the targets in the most efficient manner. It is difficult to use load shedding with mining algorithms because the stream which was dropped might represent a pattern of interest in time series analysis. The problems found in sampling are even present in this technique also. Still it had been worked on sliding window aggregate queries.

2.3 Sketching
Sketching [1, 3] is the process of randomly projecting subset of the features. It is the process of vertical sampling the incoming stream. Sketching has been applied in comparing different data streams and in aggregate queries. The major drawback of sketching is that of accuracy because of which it is hard to use this technique in data stream mining. Techniques based on sketching are very convenient for distributed computation over multiple streams. Principal Component Analysis (PCA) would be a better solution if being applied in streaming applications.

2.4 Synopsis Data Structures
Synopsis data structures hold summary information over data streams. It embodies the idea of small space, approximate solution to massive data set problems. Construction of summary or synopsis data structures over data streams has been of much interest recently. Complexities calculated cannot be O (N) where N can be space/time cost per element to solve a problem. Some solution which gives results closer to O (poly (log N)) is needed. Synopsis data structures are developed which use much smaller space of order O (logk N). These structures refer to the process of applying summarization techniques.The smaller space which contains summarized information about the streams is used for gaining knowledge.
There are a variety of techniques used for construction of synopsis data structures. These methods are briefly discussed.
2.4.1 Sampling methods:
This method is easy to use with a wide variety of applications as it uses the same multi-dimensional representation as the original data points. In particular reservoir based sampling methods were very useful for data streams.
2.4.2 Histograms:
Another key method for data summarization is that of histograms. Approximate the data in one or more attributes of a relation by grouping attribute values into “buckets” (subsets) and approximating true attribute values and their frequencies in the data based on a summary statistic maintained in each bucket [3]. Histograms have been used widely to capture data distribution, to represent the data by a small number of step functions. These methods are widely used for static data sets. However most traditional algorithms on static data sets require super-linear time and space. This is because of the use of dynamic programming techniques for optimal histogram construction. For most real-world databases, there exist histograms that produce low-error estimates while occupying reasonably small space. Their extension to the data stream case is a challenging task.
2.4.3 Wavelets:
Wavelets [11] are a well known technique which is often used in databases for hierarchical data decomposition and summarization. Wavelet coefficients are projections of the given set of data values onto an orthogonal set of basis vector. The basic idea in the wavelet technique is to create decomposition of the data characteristics into the asset of wavelet functions and basic functions. The property of the wavelet method is that the higher order coefficients of the decomposition illustrate the broad trends in the data, whereas the more localized trends are captured by the lower order coefficients. In particular, the dynamic maintenance of the dominant coefficients of the wavelet representation requires some novel algorithmic techniques. There has been some research done in computing the top wavelet coefficients in the data stream model. The technique of Gilbert gave rise to an easy greedy algorithm to find the best B-term Haar wavelet representation.
2.4.4 Sketches:
Randomized version of wavelet techniques is called sketch methods. These methods are difficult to apply as it is difficult to intuitive interpretations of sketch based representations. The generalization of sketch methods to the multi-dimensional case is still an open problem.
2.4.5 Micro Cluster based summarization:
A recent micro-clustering method [11] can be used be perform synopsis construction of data streams. It uses Cluster Feature Vector (CFV) [8]. This micro-cluster summarization is applicable for the multi-dimensional case and works well to the evolution of the underlying data stream.
2.4.6 Aggregation:

Summarizations of the incoming stream are generated using mean and variance. If the input streams have highly fluctuating data distributions then this technique fails. This can be used for merging offline data with online data which was studied in [12]. It is often considered as a data rate adaptation technique in a resource-aware mining.
Many synopsis methods such as wavelets, histograms, and sketches are not easy to use for the multi-dimensional cases. The random sampling technique is often the only method of choice for high dimensional applications.

III. ALGORITHMS FOR EFFICIENT UTILIZATION OF TIME AND SPACE

The existing algorithms used for data mining are modified from generation of efficient algorithms for data streams. New algorithms have to be invented to address the computational challenges of data streams. The techniques which fall into this category are approximation algorithm, sliding window algorithm and output granularity algorithm.
We examine each of these techniques in the context of analyzing data streams.

3.1 Approximation algorithm

Approximation techniques are used for algorithm design. The solutions obtained by this algorithm are approximate and are error bound. These have attracted researchers as a direct solution for data streams. Data rates with the available resources cannot be solved. To provide absolution to these other tools should be used along with this algorithm. For tracking most frequent items dynamically this algorithm was used in order to adapt to the available resources [16].

3.2 Sliding Window

In order to analysis recent data, sliding window protocol is used which is considered as an advanced technique for producing approximate answers to a data stream query. Analysis over the new arrived data is done using summarized versions of previous data. This idea has been adopted in many techniques in the undergoing comprehensive data stream mining system *MAIDS*. Using sliding window protocol the old items are removed and replaced with new data streams. Two types of windows called count-based windows and time-based windows are used. Using count-based windows the latest N items are stored. Using the other type of window we can store only those items which have been generated or have arrived in the last T units of time. As it emphasizes recent data, which in the majority of real-world applications is more important and relevant than older data.

3.3 Algorithm Output Granularity (AOG)

AOG is the first resource-aware data analysis approach used with fluctuating very high data rates. It works well with available memory and with time constraints. The stages in this process are mining data streams, adaptation of resources and streams and merging the generated structures when running out of memory. These algorithms are used in association, clustering and classification.

IV. APPLICATION OF METHODOLOGIES FOR STREAM MINING

The need to understand the enormous amount of data being generated every day in a timely fashion has given rise to a new data processing model-data stream processing. In this new model, data arrives in the form of continuous, high volume, fast and time-varying streams and the processing of such streams entail a near real-time constraint. The algorithmic ideas above presented have proved powerful for solving a variety of problems in data streams. A number of algorithms for extraction of knowledge from data streams were proposed in the domains of clustering, classification and association. In this section, an overview on mining these streams of data are presented.

4.1 Clustering

Guha et al. [7] Have studied analytically clustering data streams using the K - median technique. The proposed algorithm makes a single pass over the data stream and uses small space. It requires O (nk) time and O (n E) space where "k" is the number of centers, "n" is the number of points and E <1.They have proved that any k-median algorithm that achieves constant factor approximation cannot achieve a better runtime than O (nk). The algorithm starts by clustering a calculated size sample according to the available memory into 2k, and then at a second level, the algorithm clusters the above points for a number of samples into 2k and this process is repeated to a number of levels, and finally it clusters the 2k clusters into k clusters.
The exponential histogram (EH) data structure and another k-median algorithm that overcomes the problem of increasing approximation factors in the Guha et al [7] algorithm. Another algorithm that captured the attention of many scientists is the k-means clustering algorithm.
Domingos et al. [13] have proposed a general method for scaling up machine learning algorithms. They have termed this approach Very Fast Machine Learning VFML. This method depends on determining an upper bound for the Leamer's loss as a function in a number of data items to be examined in each step of the algorithm. They have applied this method to K-means clustering VFKM and decision

tree classification VFDT techniques. These algorithms have been implemented and evaluated using synthetic data sets as wells real web data streams. VFKM uses the Hoeffding bound to determine the number of examples needed in each step of K-means algorithm. The VFKM runs as a sequence of K-means executions with each run uses more examples than the previous one until a calculated statistical bound (Hoeffding bound) is satisfied.

A fast and effective scheme that is capable of incrementally clustering moving objects. They used notion of object dissimilarity, which is capable of taking the future object movement which improves the quality clustering and runtime performance. The Experiments conducted show that the proposed MC algorithm is 106 times faster than K-Means and 105 times faster than Birch. They have used average-radius function which automatically detects cluster split events.

Ordonez [6] has proposed an improved incremental k-means algorithm for clustering binary data streams. O'Challaghan et al. [4] proposed STREAM and LOCALSEARCH algorithms for high quality data stream clustering. Aggarwal et al. [11] have proposed a framework for clustering evolving data steams called CluStream algorithm. In [12] they have proposed the HPStream; a projected clustering for high dimensional data streams, which has outperformed CluStream. Stanford's STREAM project has studied the approximate k-median clustering with guaranteed probabilistic bound.

4.2 Classifications

In the real world concepts are often not stable but change with time. Typical examples of this are weather prediction rules and customers' preferences. The underlying data distribution may change as well. Often these changes make the model built on old data inconsistent with the new data and regular updating of the model is necessary. This problem is known as concept drift.

Wang et al. [14] proposed a general framework for mining concept drifting data streams. This was the first framework which worked on drifting. They have used a weighted classifier to mine streams and old data expires based on the distribution of data. The proposed algorithm combines multiple classifiers weighted by their expected prediction accuracy. Lastly proposed an online classification system which dynamically adjusts the size of the training window and the number of new examples between model reconstructions to the current rate of concept drift.

Domingos et al. [15] have developed Vary Fast Decision Tree (VFDT) which is a decision tree constructed on Hoeffding trees. The split point is found by using Hoeffding bound which satisfies the statistical measure. This algorithm also drops the non-potential attributes. Aggarwal has used the idea of micro clusters introduced in CluStream in On-Demand classification and it shows a high accuracy. The technique uses clustering results to classify data using statistics of class distribution in each cluster.

Peng Zhang [10] have proposed a solution for concept drifting where in the categorized the concept drifting in data streams into two scenarios: (1) Loose Concept Drifting (LCD); and (2) Rigorous Concept Drifting (RCD), and proposed solutions to handle each of them by using weighted-instancing and weighted classifier ensemble framework such that the overall accuracy of the classifier ensemble built on them can reach the minimum. Ganti et al. [10] have developed analytically two algorithms GEMM and FOCUS for model maintenance and change detection between two data sets in terms of the data mining results they induce.

4.3 Association

The approach proposed and implemented an approximate frequency count in data streams which uses all the previous historical data to calculate the frequency patterns incrementally. Cormode and Muthukrishnan [16] developed an algorithm for counting frequent items. This is used to approximately count the most frequent items. BIDE was proposed which efficiently mines frequent items without candidate maintenance. It uses a novel sequence called BIDirectional Extension and prunes the search space more deeply than the previous algorithm. The experimental results show that BIDE algorithm uses less memory and faster when support is not greater than 88 percent. An algorithm called DSP which efficiently mines frequent items in the limited memory environment. The algorithm which focuses on mining process; discovers from data streams all those frequent patterns that satisfy the user constraints, and handle situations where the available memory space is limited.

4.4 Frequency Counting

Frequency counting has not much in attention among the researchers in this field, as did clustering and classification. Counting frequent items or itemsets is one of the issues considered in frequency counting. Cormode and Muthukrishnan [16] have developed an algorithm for counting frequent items. The algorithm maintains a small space data structure that monitors the transactions on the relation, and when required, quickly outputs all hot items, without rescanning the relation in the database. Even had developed a frequent item set mining algorithm over data stream. They have proposed the use of tilted windows to calculate the frequent patterns for the most recent transactions. The implemented an approximate frequency count in data streams. The implemented

algorithm uses all the previous historical data to calculate the frequent pattern incrementally. One more AOG-based algorithm: Lightweight frequency counting *LWF*. It has the ability to find an approximate solution to the most frequent items in the incoming stream using adaptation and releasing the least frequent items regularly in order to count the more frequent ones.

4.5 Time Series Analysis

Time series analysis had approximate solutions for the error bounding problems. The algorithms based on sketching techniques process of starts with computing the sketches over an arbitrarily chosen time window and creating what so called sketch pool. Using this pool of sketches, relaxed periods and average trends are computed and were efficient in running time and accuracy.

Then have proposed a two phase approach to mine astronomical time series streams. The first phase clusters sliding window patterns of each time series. Using the created clusters, an association rule discovery technique is used to create affinity analysis results among the created clusters of time series. The proposed techniques to compute some statistical measures overtime series data streams. The proposed techniques use discrete Fourier transform. The use of symbolic representation of time series data streams. This representation allows dimensionality/numerosity reduction. They have demonstrated the applicability of the proposed representation by applying it to clustering, classification, and indexing and anomaly detection. The approach has two main stages. The first one is the transformation of time series data to Piecewise Aggregate Approximation followed by transforming the output to discrete string symbols in the second stage. The application of what so called regression cubes for data streams was proposed. Due to the success of OLAP technology in the application of static stored data, it has been proposed to use multidimensional regression analysis to create a compact cube that could be used for answering aggregate queries over the incoming streams.

The randomized variations of segmenting time series data streams generated on-board mobile phone sensors. One of the applications of clustering time series discussed: Changing the user interface of mobile phone screen according to the user context. It has been proven in this study that Global Iterative Replacement provides approximately an optimal solution with high efficiency in running time.

V. RESEARCH ISSUES

The study of Data stream mining has raised many challenges and research issues

• Optimize the memory space, computation power while processing large data sets as many real data streams are irregular in their rate of arrival, exhibiting burstiness and variation of data arrival rate over time.

• Variants in mining tasks those are desirable in data streams and their integration.

• High accuracy in the results generated while dealing with continuous streams of data

• Transferring data mining results over a wireless network with a limited bandwidth.

• The needs of real world applications, even mobile devices

• Efficiently store the stream data with timeline and efficiently retrieve them during a certain time interval in response to user queries is another important issue.

• Online Interactive processing is needed which helps user to modify the parameters during processing period.

VI. CONCLUSIONS

The dissemination of data stream phenomenon has necessitated the development of stream mining algorithms. So in this paper we discussed the several issues that are to be considered when designing and implementing the data stream mining technique. We even reviewed some of these methodologies with the existing algorithm like clustering, classification, frequency counting and time series analysis have been developed.

We can conclude that most of the current mining approaches use one passes mining algorithms and few of them even address the problem of drifting. The present techniques produce approximate results due to limited memory.

Research in data streams is still in its early stage. Systems have been implemented using these techniques in real applications. If the problems are addressed or solved and if more efficient and user-friendly mining techniques are developed for the end users, it is likely that in the near future data stream mining will play an important role in the business world as the data flows continuously.

REFERENCES

[1]. S. Muthukrishnan (2003), Data streams: algorithms and applications, Proceedings of the fourteenth annual ACM- SIAM symposium on discrete algorithms.

[2]. P. Domingos and G. Hulten, A General Method for Scaling Up Machine Learning Algorithms and its Application to Clustering, Proceedings of the Eighteenth International Conference on Machine Learning, 2001, Williamstown, MA, Morgan Kaufmann.

[3]. B. Babcock, S. Babu, M Datar, R Motwani, and 1. Widom, Models and issues in data stream systems, in Proceedings of PODS, 2002.

[4] L. O'Callaghan, N. Mishra, A. Meyerson, S. Guha, and R. Motwani. Streaming-data algorithms for high-quality clustering. Proceedings of IEEE International Conference on Data Engineering, March 2002.
[5]. Aggarwal C, Han 1., Wang 1., Yu P. (2003) A Framework for Clustering Evolving Data Streams, VLDB Conference.
[6] C. Ordonez. Clustering Binary Data Streams with K-means ACM DMKD 2003.
[7]. S. Guha, N. Mishra, R Motwani, and L. O'Callaghan. Clustering data streams. In Proceedings of the Annual Symposium on Foundations of Computer Science, IEEE, November 2000.
[8]. Zhang, T., Ramakrishnan, R, Livny, M.: Birch: An efficient data clustering method for very large databases, In: SIGMOD, Montreal, Canada, ACM (1996).
[9] C. Aggarwal, J. Han, J. Wang, and P. S. Yu, On Demand Classification of Data Streams, Proc. 2004 Int. Conf. on Knowledge Discovery and Data Mining, Seattle, WA, Aug. 2004.
[10] V. Ganti, J. Gehrke, and R. Ramakrishnan: Mining Data Streams under Block Evolution. SIGKDD Explorations 3(2), 2002.
[11]. Keirn D. A. Heczko M. (2001) Wavelets and their Applications in Databases. ICDE Conference.
[12]. C Aggarwal, J. Han, J. Wang, and P. S. Yu, A Framework for Projected Clustering of High Dimensional Data Streams, Proc. 2004Int. Conf on Very Large Data Bases, Toronto, Canada, 2004.
[13]. G. Hulten, L. Spencer, and P. Domingos. Mining Time-Changing Data Streams. ACM SIGKDD 2001.
[14]. H. Wang, W. Fan, P. Yu and 1. Han, Mining Concept-Drifting Data Streams using Ensemble Classifiers, in the 9th ACM International Conference on Knowledge Discovery and Data Mining (SIGKDD), Aug. 2003, Washington DC, USA
[15]. P. Domingos and G. Hulten. Mining High-Speed Data Streams. In Proceedings of the Association for Computing Machinery Sixth International Conference on Knowledge Discovery and Data Mining, 2000.
[16]. G. Cormode, S. Muthukrishnan What's hot and what's not: tracking most frequent items dynamically. PODS 2003: 296-306