ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 3, March 2016

A Survey on New Techniques Over Data Streams

Priyadarshini.I.Kuchanur

M.E Student, Dept. of C.E., SVIT, Vasad, Anand, India

ABSTRACT: The present progresses in hardware and software have empowered the capture of different amounts of data in a wide range of fields. These amounts are generated continuously and in a very high fluctuating data rates. Examples include sensor networks, web logs, and computer network traffic. The efficient storage of data, querying with its optimization and mining of such data sets are highly computationally challenging tasks. Mining data streams is concerned with mining knowledge structures signified in models and patterns in non stopping streams of information. The research in data stream mining has drawn-out a high attraction due to the significance of its applications and the increasing generation of streaming information. Applications of data stream analysis in real world can differ from critical scientific and astronomical applications to important business and commercial finances ones. In this review paper, we present the state of- the-art in this growing dynamic field.

KEYWORDS: Data Streams; Clustering; Classification; Frequent Mining; Change Diagnosis; Query Estimation

I. INTRODUCTION

"Data mining is computational process of discovering patterns in large data sets." The intelligent data analysis has passed through a number of stages. Each stage addresses novel research issues that have arisen. Statistical exploratory data analysis represents the first stage. The goal was to explore the available data in order to test a specific hypothesis. [12] It involves methods interacting with artificial intelligence, database systems, statistics and machine learning. Goal of data mining process is to extract information from data sets and to make an understandable structure for further use. It's a process of transforming raw data into understandable structure.

Recently a new class of emerging applications has become widely recognized: applications in which data is generated at very high rates in the form of transient *data streams*. Examples of such applications include financial applications, network monitoring, security, telecommunication data management, web applications, manufacturing, sensor networks, and others. In the data stream model, individual data items may be relational tuples, e.g., network measurements, call records, web page visits, sensor readings, and so on. However, their continuous arrival in multiple, rapid, time-varying, possibly unpredictable and unbound streams open new fundamental research problems. This speedy generation of continuous streams of information has challenged our storage, computation and communication abilities in computing systems. The vast amounts of data arriving in high speeds need employment of semi-automated interactive techniques, to perform real-time extraction of hidden knowledge and information.

Stream data has become a challenge to today's analytic processes in Knowledge Discovery and Data mining (KDD) due to its large size and dynamic characteristics when being generated. Moreover, various problems in managing and processing of stream data are also issued from high-dimensional attributes and multi-valued categorical values found in recent practical data. Therefore, an ideal analytic task must catch two goals; it should not only preserve sufficient effectiveness, but also be efficient to load, analyze, and process queries continuously and simultaneously with these massive volumes of information exploding. Streaming Data has following characteristics like data arrives continuously, ordering assumptions cannot be made in data streams and length of stream data is unbounded. It is difficult to capture information from data streams efficiently and effectively.

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 3, March 2016

II. METHODOLOGIES

Mining data streams has gathered the attention of data mining community for the last one decade. A number of algorithms have been proposed for extracting knowledge from streaming information. In this section, we review Clustering, Classification, Frequency Counting, Change Diagnosis and Query Estimation are discussed.

A. Data Clustering

Data stream clustering by example treats data points coming at same time stamp from different data source as one unit in clustering process. Each data unit describes the features of an entity at a particular time stamp. For an example in a bank daily, there are many transaction that are on different account numbers, are of different type like credit or debit by cash, check or online transaction and also other features are associated like avail balance, amount deposited or withdrawn etc. so for a single transaction in this case many data points are recoded that are of different types and in combination form a data unit. New clustering algorithms are discussed as under:

- STREAM [4] is clustering algorithm which consists of two phases divide and conquer approach. In first phase that is divide phase which divides data streams in many buckets and then finds k clusters in each bucket y applying k – median clustering. It stores cluster centres only. These cluster centres are weighted based on number of data points belongs to corresponding cluster and the discard data points. In second phase weighted cluster centres are clustered in small number of clusters. It cannot adapt to concept evolution in data. It has low space and time complexity.
- CluStream [5] divides clustering process in two components online component and offline components. Online component stores summary of data in form of micro – clusters. Summary statics of data are stored in snapshots form which provides the user flexibility to specify the time interval for clustering of micro - clusters .Offline component apply the k – means clustering to cluster micro- clusters in to bigger cluster.
- HPStream [6] is clustering for high dimensional data streams. It uses a Fading Cluster Structure to stores summary of streaming data. It gives importance to recent data by fading the old data with respect to change in time. To handle high dimensional data it selects subset of dimensions by cast on original high dimensional data stream, number of dimensions and dimensions are not same for each cluster. This is due to relevance of each dimension in each cluster may differ, its incrementally updatable and highly scalable on number of dimensions. It cannot discover the cluster of arbitrary shapes and require domain knowledge for specifying number of clusters and average number of projected dimensions parameters.
- DenStream [7] is density based clustering algorithm. Concept of core point of DBSCAN is extended and notion of micro cluster is employed to store an approximate representation of data points. DenStream works in two phase, one is online phase and other is offline phase. Online phase maintains micro- cluster structure and offline phase generates final clusters from the set of Online maintained micro clusters by applying a variant of DBSCAN algorithm which is according to demand by user. It also facilitates to handle outliers. It improves accuracy by giving second chance to dropped clusters.
- D-stream [8] is density based grid clustering algorithm for continuous stream data. It consists of two phases online phase and offline phase. It divides whole data in grids. In online phase it maps incoming data point on corresponding grid. In offline phase it calculates density of each grid and then discard the data. To decrease the density with time it uses fading function, if it falls below threshold and no new data point is added since last checking of grid density that grid is discarded. There is exponential relation between number of dimensions and number of grids, this makes it non scalable.
- E- Stream [9] is clustering technique which supports five types of evolution in streaming data: Arrival of new cluster, Split of a large cluster, union of two similar clusters, disappearance of an old cluster and variation in behaviour of cluster itself. Fading cluster structure is used with histogram to approximate the streaming data. It requires more parameters specified by user.
- DUCstream (Dense Units Clustering for data stream) [10] is grid based technique which splits the data space in the non-overlapping grids and process's data in form of chunks. The chunk is consists of M number of data

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 3, March 2016

points such that it could fit in main memory. Data points are mapped in corresponding grid and then forms clusters by applying depth first search algorithm. It identifies the clusters as a connected components of the graph whose vertices represent dense units and whose edges correspond to the mutual faces between two vertices.

B. *Data Classification*

Classification is a two step process in which it initially learns from training data to form a classifier which is then used to classify unknown samples from testing data. The stream classifier must evolve to effectively indicate current class distribution in case of evolving data streams [11]. There are two widely used classification approaches:

- VFDT [12, 13] is approach which uses decision tree learning and hoeffding bounds to guarantee approximate correct output. Since data is stationary it can be process in stable time and space. This algorithm does not requires to store examples in memory and can learn by seeing each example once. The drawback of VFDT is improvised in CVFDT that addresses the concept drift problem in time varying data streams. CVFDT performance is same as VFDT in terms of speed and accuracy. CVFDT detects and responds to the changes that might occur in data without constructing a target classification model from scratch. CVFDT provides easy way of handling concept drift by inclusion of alternate sub trees.
- Ensemble Classifier [14] boosts classification accuracy. The idea of ensemble is to train group of classifiers with different chunks of data stream. For new grown trees is trained by different chunks or windows of data. For new records each tree produces n different predictions. The majority voted class is considered to be prediction of whole ensemble. Each individual tree prediction is assigned weight on the basis of accuracy in time varying environment. The decision is made on highest majority votes of selected k classifiers. In order to overcome drawback of weighted ensemble classifier method, multi chunk partition in multi ensemble method was developed. Multi ensemble cuts error rate over single partition-single chunk which uses simple majority voting.

C. F*requent Item Set Mining*

Associated to traditional databases, mining in data streams has more limitations and requirements. First, each element (e.g., transaction) in the data stream can be examined just the once or twice, making traditional multiple-scan approaches infeasible. Second, the usage of memory space should be limited in a range, despite that data elements are continuously streaming into the local site. The mining task should proceed normally and offer acceptable quality of results. Fourth, the latest analysis result of the data stream should be available as soon as possible when the user invokes a query. This result, one good stream mining algorithm to possess efficient performance and high throughput. Some of such algorithms are discussed as under.[15]

- Gwangbumet [15] suggested an improved tree structure to implement an outstanding frequent mining technique called Linear Prefix Tree (LP-Tree). LP Tree is composed of array forms to decrease pointers between nodes. In LP Tree separate arrays are used to store each transaction. Tree Structure of LP Tree is as follows:

  LP-tree = {Headerlist,BNL,LPN1,LPN2,…,LPNn}

  BNL is an another extension to keep track of the branched node. Header list is same as FP tree, it contains sorted items, items frequency and pointer to pointing to the first node of the specific item. The process is starting by scanning the database to find the frequency of the each item, then create the header list and sort it in descending order of the frequency. In second scan each transaction is being considered, prune it by removing infrequent items and sort in descending order of the frequency. The length of the array is equal to the number of items in the proceed transaction. The processed transaction will be stored in newly created LPN which is combination of arrays.
- Yuh-JiuanTsay [16] suggested a novel method called Frequent Items Ultra metric trees (FIUT) for mining frequent itemset. It consists two phases, in first phase two scans of database are conducting. In first scan the frequency of all 1-itemset is calculated. Second scan prunes the transactions (delete infrequent items), counts the number of remaining items in each transactions and grouping the transactions into separate clusters based on the number of items remaining in the transaction after pruning. In second phase transactions in the clusters are considered,

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 3, March 2016

repetitive construction of FIU- tree and mining process is conducted with it. The group of transactions with highest number of items are considered first to create the k – FIU tree where k is from M down to 2 and M denotes the maximal value of k among the transactions. Transactions with m items are used to begin the construction of the tree and mine the frequent m- itemsets.

- Ke-Chung Lin [17] proposed improved frequent pattern (IFP) growth method for mining association rules. IFPgrowth employs an address- table structure to lower the complexity of searching in each node in an FP- tree. Address table is an effective data structure which contains a set of items and pointers. The pointer of an item points to the parallel node of that item at the next level of FP – tree. This address table can be used to find the child node in the FP tree easily. Each node checks its address – table to confirm whether its child exists in the tree. Each node has its own address table except the node of last item in fp tree. In address table of root includes all the frequent items and pointers to actual node in first level. At second level all nodes include a separate address table with all items which are came after the current item in the header table. Same address table may be repeated as items may appear more than once in fp, which consumes huge amount of memory.

Frequent pattern mining using windowing technique which are transaction sensitive. This windowing method mines complete set of repeated items and uses bit presentation for existence and absence as 1 and 0's respectively. This works in three phases first phase refers to windows initialization and obtains bit sequence of any data items which is equal to window length. Second is window sliding phase old transactions are deleted and new one are added and sliding is been shifted to left in bit sequence. Third phase discovers frequent pattern.

- Eclat [18] algorithm works on depth first algorithm to find repeated items which is functioning on vertical view. Database is in list form which keeps vertical view in transactions in form of tid-list. Then support of any of data set is found by intersecting on these lists and assumed tree is made by depth first search and repeated data are found.
- LDS [19] algorithm overcomes the deficiency of Eclat that it is not suitable for dispersed data items. LDS keeps list of transaction numbers as its data items and if length of list is longer than half of the window length then its complementary list is kept instead of usual one. Here complementary means list of transactions numbers in the window where they do not exists on usual list of respective data item.
- MFPN algorithm performs window upgrading based on pin. Transactions are numbered in each pin and following the addition to new pin, operations to make transactions list in vertical manner with pin numbers in the window. Simultaneously with every new pin algorithm performs mining and finds set of repeated data items and results are displayed based on user request.

D. Change Diagnosis Algorithm

Change is an underlying phenomenon. Data will keep on changing with time passes which creates more data with different amount of results which increments in record storage. This process is called as data evolution. It is necessary to track changes in data and as soon as possible. The board classification of change diagnosis algorithms are as Velocity Density Estimation and Windowing Based Techniques.

- Velocity Density Estimation [20] computes rate of change in data density of different data points in data streams over time. Depending on rate of change with its density and its direction, one may identify regions of dissolution, coagulation and shift. Directions of shifts can be constructed by spatial profiles of underlying data. In high level evolution, it is possible to identify combinations of dimensions with use of velocity density concept.

- Window based techniques [21] uses window to handle concept drift or change in data. A window is a short memory data structure which can store set of useful data sample or summarizes some statistics concerning model behaviour or data distribution in order to characterize the current concept. Windows can be data based or time based. In data based window size is characterized by number of instances or data samples. Data based windows can be called as sequence based window. In time based window, window size is defined by duration or period of

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 3, March 2016

time. Window based change diagnosis can be characterized as single window, two window and multiple window based processing.

In single window based processing, periodically updates data sample stored in one window. If window is of fix size then it stores data in FIFO (First In First Out) manner. There is increase in size of window when there is no change and once change is detected window size is reduced by containing most representative data samples.
In two window based processing, two windows are used. First window is referenced window to represent initial concept and second window is to represent current concept. These two windows can be separated, adjacent, overlapped and discontinued.

- Separated windows were reference and current windows are departed as data is processed. Separated windows are most helpful in handling gradual continuous drift. If performance is degrade below certain threshold then learner is relearned from the current data batch.
- Adjacent window were reference and current windows are kept related during incoming data samples processing. Reference window and current window are sequential processed, for this onIine learner is required. Different variants are possible according to different window sizes like both windows fixed size, both variable size and one fix and variable size.
- Overlapped window both reference and current windows will have data samples in common. This is used with limited sample data size or when data set is imbalanced. This window are generally used in ensemble learners for accurately training individual learners.
- Discontinued window where only subset of reference window are used in comparison with the current window. It selects subsets of data samples from the reference window according to some criteria. After first comparison it compares them to current window. The subsets can be selected according to similarity in space, in time or in data samples which represents encountered drift. This technique is useful where drift affects a little amount of data samples in window.

E. Query Estimation in Data Streams

Query estimation problem is important for optimizing queries. These queries are most to resolve in online time. Effectiveness of query optimization depends on the different execution plans and its system’s ability. For proper execution size and data distributions of intermediate results generated during plan implementation need to be estimated accurately. Huge volume of data is over flowing continuously, it is not possible to process the data efficiently by multiple passes. Similar with data streams it is not possible to store whole data in main memory and thus summary structures are used to compute approximate answers.

- Sampling approach uses randomly samples of tuples and scales down copy of original data and estimate the results at run time. With adequate samples, can provide accurate estimation as information collected is most recent one. The main goal of sampling is to obtain highest possible accuracy with number of samples. Sampling techniques do not need storage to store statically summary.

  Chen [22] gave an approach for estimating records selective in database quires. It approximates attribute value distribution using query feedbacks. It uses subsequent query feedbacks to regress the distribution for improving accuracy. Acharya el al. computes on join synopsis for each relation for answering queries with foreign key joins. These joins for each relations are created by joining the sample of relations with other relation. Von Neuman gave golden rule for sampling cumulative probability distributions to discrete domains. Its samples are based on frequency distribution of the cumulative frequency distribution. This is applicable only on single attribute queries not on join or multidimensional.

- Histograms is popular way to approximate a data distribution. It is based on approach to divide attribute values to buckets, store information that can summarize the distribution of attribute values. There is an assumption regarding constructing histograms that data set to be approximated is finite and the size can be easily derived by performing a single pass over finite data set.

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 3, March 2016

Guha [23] gave histogram algorithm in which data streams are passed single time for fixed window data stream model which also supports incremental histograms. Gunopulos[24] addressed the problem of estimating the selectivity of multidimensional range queries for the data sets have numerical attributes with real values. The technique defines buckets of variable size and allows buckets to overlap, overlapping allows more compact approximation of the data distribution. nLT is one more query estimation algorithm which based on non bucket based histogram. The approximate value of cumulative frequency is stored in 7 intervals inside bucket. The frequency values are organised in 4 level tree over the bucket, they are called as 4 level tree (4LT) index. The 4LT tries to provide best frequency estimation inside a bucket. nLT extends idea of 4LT, it constructs entire histogram besides single bucket. It is based on hierarchical decomposition of original data in full binary tree. Tree contains pre computed hierarchical queries, this reduces space storage and increases accuracy.

- Wavelets transformations are mathematical transformations that attempt to capture the trend in numerical functions by decomposing them. Different type of functions of interest such as image, curve or surface are represented well with help of wavelets and in space efficient manner.

Matias [25] proposed a technique based on multi resolution wavelet decomposition for building histograms. Histograms are built on on-line selective estimation with limited space usage. Wavelet approximation is more effective for selectivity estimate of range queries. Wavelet based histograms using linear bases perform well on the most of query sets and data distributions. These can be extended to numerous attributes by multidimensional wavelet decomposition and reform. It requires large space to calculate wavelet coefficients in dynamic streaming environment. When data item changes in values, many coefficients may get affect and change and set of significant coefficients could change quite extensively.

III. CONCLUSION

From the above analysis of the different types of algorithms I conclude that the different algorithms for different techniques have their own criteria and own prior knowledge of the data. The different algorithms analysis shown in the tables as below:

Table 1. Comparative Analysis of Clustering Methods based on various Parameters

Sr No.	Clustering Method	Handle Concept Progression	Handle Data Fading	Clustering Tactic	Constraints
1	STREAM	No	No	Partitioning	Number of Clusters
2	CluStream	Yes	No	Partitioning & Hierarchical	Number of Clusters, Time Window
3	HPStream	Yes	Yes	Partitioning & Hierarchical	Max. Number of clusters, Average number of projection dimension
4	DenStream	Yes	Yes	Density Based	Cluster Radius Threshold, Data fading rate
5	D-stream	Yes	Yes	Density Based & Grid based	Threshold of density of grid cells, Data fading rate
6	E- Stream	Yes	Yes	Hierarchical	Max. Number of clusters, Data fading rate and 4 other parameters
7	DUCstream	Yes	No	Density Based & Grid based	Threshold of density of grid cells

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 3, March 2016

Table 2. Comparative Analysis of Classification Methods based on various Parameters

Sr No.	Algorithm	Time Complexity	Space Complexity	Handles Concept Drift
1	VFDT	Less	Less	Yes
2	CVDFT	Less	More	No
3	Weighted Ensemble	More	More	Yes
4	Multi Ensemble	More	Less	Yes

Table 3. Comparative Analysis of Query Estimation Methods based on various Parameters

Sr. No.	Category of Technique	Elementary Knowledge	Merits	Demerits
1	Sampling	It uses random samples of tuples as a knowingly scaled down copy of data	Easy to implement and does not require storage of statistical information.	Faces difficulties under updates, complex queries and multidimensional data
2	Histogram	It partitions attribute values into buckets to summarize the data distribution	Fix amount of space is used to approximate data distribution.	Faces difficulties under updates and multidimensional data. It is not easy to handle large domain of attributes.
3	Wavelets	This method attempts to capture broad trends in data by decomposing data into more significant coefficient.	Requires small number of significant coefficients for capturing trends in numerical function and can be applied to multiple attributes.	Needs large space under dynamic environment. Updating is difficult and cannot be applied directly to data stream processing.

References

1. Yogita, Y., and D. Toshniwal. "Clustering techniques for streaming data-a survey." *Advance Computing Conference (IACC), 2013 IEEE 3rd International*. IEEE, 2013.
2. Jamsheela, O., and G. Raju. "Frequent itemset mining algorithms: A literature survey." *Advance Computing Conference (IACC), 2015 IEEE International*. IEEE, 2015.
3. Gupta, Sudhanshu, and Deepak Garg. "Survey on query estimation in data streams." *2009 IEEE International Advance Computing Conference*. 2009.
4. O'callaghan, Liadan, et al. "Streaming-data algorithms for high-quality clustering." *icde*. IEEE, 2002.
5. Aggarwal, Charu C. "A framework for diagnosing changes in evolving data streams." *Proceedings of the 2003 ACM SIGMOD international conference on Management of data*. ACM, 2003.
6. Aggarwal, Charu C., et al. "A framework for projected clustering of high dimensional data streams." *Proceedings of the Thirtieth international conference on Very large data bases-Volume 30*. VLDB Endowment, 2004.
7. Cao, Feng, et al. "Density-Based Clustering over an Evolving Data Stream with Noise." *SDM*. Vol. 6. 2006.
8. Li-xiong, Liu, et al. "rDenStream, A Clustering Algorithm over an Evolving Data Stream." *Information Engineering and Computer Science, 2009. ICIECS 2009. International Conference on*. IEEE, 2009.

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 3, March 2016

9. Udommanetanakit, Komkrit, ThanawinRakthanmanon, and KitsanaWaiyamai. "E-stream: Evolution-based technique for stream clustering."*Advanced Data Mining and Applications*. Springer Berlin Heidelberg, 2007. 605-615.
10. Gao, Jing, et al. "An incremental data stream clustering algorithm based on dense units detection." *Advances in Knowledge Discovery and Data Mining*. Springer Berlin Heidelberg, 2005. 420-425.
11. Attar, Vahida, Pradeep Sinha, and Kapil Wankhade. "A fast and light classifier for data streams." *Evolving Systems* 1.3 (2010): 199-207.
12. Domingos, Pedro, and Geoff Hulten. "Mining high-speed data streams."*Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2000.
13. Hulten, Geoff, Laurie Spencer, and Pedro Domingos. "Mining time-changing data streams." *Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2001.
14. Street, W. Nick, and YongSeog Kim. "A streaming ensemble algorithm (SEA) for large-scale classification." *Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2001.
15. Pyun, Gwangbum, Unil Yun, and Keun Ho Ryu. "Efficient frequent pattern mining based on linear prefix tree." *Knowledge-Based Systems* 55 (2014): 125-139.
16. Tsay, Yuh-Jiuan, Tain-Jung Hsu, and Jing-Rung Yu. "FIUT: A new method for mining frequent itemsets." *Information Sciences* 179.11 (2009): 1724-1737.
17. Lin, Ke-Chung, I-En Liao, and Zhi-Sheng Chen. "An improved frequent pattern growth method for mining association rules." *Expert Systems with Applications* 38.5 (2011): 5154-5161.
18. Zaki, Mohammed Javeed, et al. "New Algorithms for Fast Discovery of Association Rules." *KDD*. Vol. 97. 1997.
19. Deypir, Mahmood, Mohammad HadiSadreddini, and Mehran Tarahomi. "An Efficient Sliding Window Based Algorithm for Adaptive Frequent Itemset Mining over Data Streams." *J. Inf. Sci. Eng.* 29.5 (2013): 1001-1020.
20. Aggarwal, Charu C. *Data streams: models and algorithms*. Vol. 31. Springer Science & Business Media, 2007.
21. “Change detection in streaming data” by M.SC. Dang Hoan Tran
22. Chen, Chungmin Melvin, and Nick Roussopoulos. *Adaptive selectivity estimation using query feedback*. Vol. 23. No. 2. ACM, 1994.
23. Ganguly, Sumit, et al. "Join-distinct aggregate estimation over update streams." *Proceedings of the twenty-fourth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*. ACM, 2005.
24. Gunopulos, Dimitrios, et al. "Selectivity estimators for multidimensional range queries over real attributes." *The VLDB Journal—The International Journal on Very Large Data Bases* 14.2 (2005): 137-154.
25. Jiang, Zhewei, et al. "Join size estimation over data streams using cosine series." *Int J InfTechnol* 12.9 (2007): 27-45.

BIOGRAPHY

Priyadarshini I Kuchanuris a M.E student in the Computer Engineering Department, SardarVallabdhbhai Patel Institute of Technology, Vasad and PursuingMaster of Engineering (ME) degree in 2016 from SVIT, Vasad, Anand, India. Research interests are Data mining (Data Stream Mining), Change Detection Algorithms, and Volatiltiy Detection through Drift Detection etc.