A Framework for On-Demand Classification of Evolving Data Streams

*Ms.Pranali R Gajbhiye ,Prof. P.D.Sathya

*ME Student , Department of Computer Science & Engineering , BAMU University, SYCET,Aurangabad.

Assistant Professor , Department of Computer Science & Engineering , BAMU University, SYCET, Aurangabad

Abstract*:*Current models of the classification problem do not effectively handle bursts of particular classes coming in at different times. In fact, the current model of the classification problem simply concentrates on methods for one-pass classification modeling of very large data sets. Our model for data stream classification views the data stream classification problem from the point of view of adynamic approach in which simultaneous training and test streams are used for dynamic classification of data sets. This model reflects real-life situations effectively, since it is desirable to classify test streams in real time over an evolving training and test stream. The aim here is to create a classification system in which the training model can adapt quickly to the changes of the underlying data stream. In order to achieve this goal, we propose an on-demand classification process which can dynamically select the appropriate window of past training data to build the classifier. The empirical results indicate that the system maintains a high classification accuracy in an evolving data stream, while providing an efficient solution to the classification task.

Keywords: Stream classification, geometric time frame, microclustering, nearest neighbor

I. INTRODUCTION

As far as real world application considers, there's a necessity in data storage technology haveled to the ability to store the data for real-timetransactions. Such processes lead to data which often grow without limit and are referred to as data streams. Discussions on recent advances in data stream mining may be found in [4].One important data mining problem which has been studied in the context of data streams is that ofclassification[10].

. We develop such an on-demand classifier. The on-demand classifier is designed by adapting the (unsupervised) microclustering model [2] to the classification problem. Since microclustering is a data summarization technique, some of the underlying conceptscan be leveraged effectively for other problems, such as classification, which utilize the aggregate data behavior over different time horizons. In order to use such an approach for the classification problem, the following adaptations need to be made:

1. The microclustering process needs to be supervised, since each microcluster belongs to a specific class. Therefore, the representation of the microclusters and the process of updating, merging, and deletingmicroclustersneeds to be done in a class-specific way. The aim of microclustering is to test the classdiscrimination of different time horizons.

2.A geometric time frame is used instead of the pyramidal time frame in order to store the supervised microclusters. We discuss the similarities and differences of these time frames, and also discuss the advantages of the geometric time frame.

3.A testing phase needs to be designed in conjunction with the creation of the supervised microclusters. This testing phase needs to be sensitive to the evolution of the underlying data stream.

4.Methods need to be designed to pick the optimumsegment of the stream in order to effectively perform the classification process. This is because the evolution of the stream [3] significantly affects the behavior of the classification algorithm. For thispurpose, the classification framework needs to divide the training stream into two parts which are discussed below.

The testing phase of the on-demand classifier is constructed by dividing the training stream into two parts:

1. A portion which is used for class-specific statistical maintenance of microclusters.
2. A portion which is used for testing the nature of the horizon which provides the best classification accuracy.

II. DEVELOPED SYSTEM

The unsupervised microclustering approach developed in [2] in order to make it work effectively for the classification problem in the context of highly evolving data streams. Recent papers have proposed the classification model in a data stream as a relatively straightforward extension of the traditional classification problem. The only difference is that one-pass mining is required in order to perform the training. In reality, the process of classification should be viewed as a continuous process in which the training stream and test stream are simultaneously generated by the underlying process.

In addition, it is assumed that both the training and test streams are evolving over time. This assumption may be true in many monitoring scenarios in which the activities inthe underlying data stream are followed by events which can be tracked in time. For example, in business activity monitoring applications, it may be possible to track variouscontrol variables as the underlying training stream and the events of significance as the test stream. The same is true of surveillance applications in which a large number of variables may be tracked in order to monitor events of significance. For applications in which manual labeling is required, this may be true if the class of interest occurs as a rare event. In such cases, the data stream may have a high volume, but the system may be augmented by periodic labeling when the rare events of interest do occur. The assumption of simultaneous test and training streams may not always be necessary when the entire training data is already available, as in the case of static databases.

However, in applications in which the classification is used as a means to a rapid response mechanism, this assumption turns out to be very useful. Such applications are also referred to as on-demand applications.

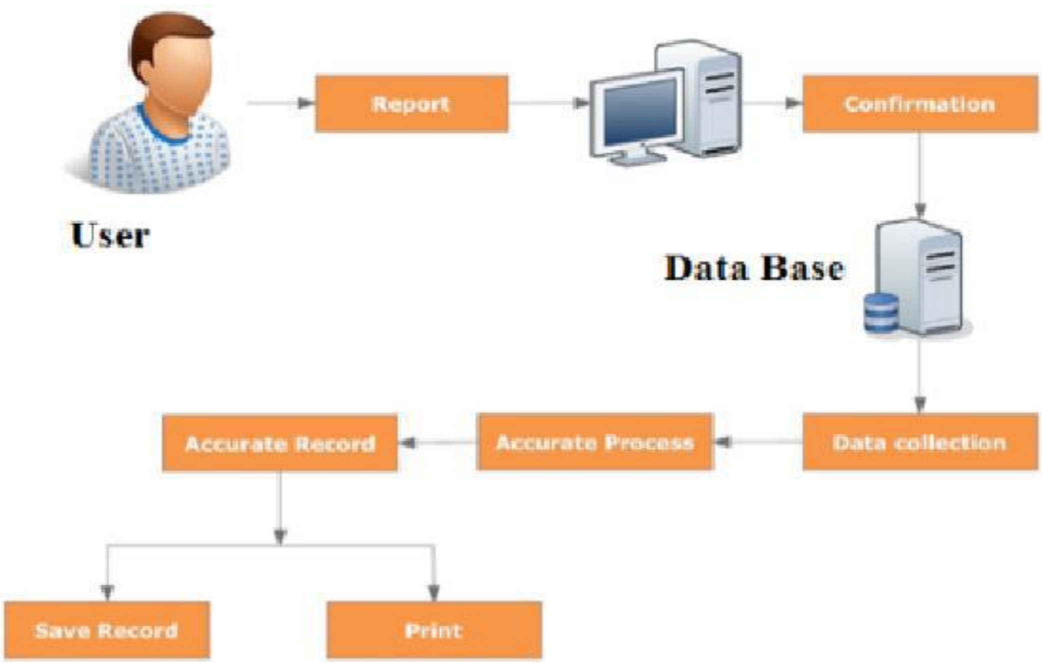

Fig 1.System Architecture

To provide users with different forms of flexibility asflexible query language as well as flexible and interactive retrieval process. By adapting the temporalabstraction [1], the most accurate record will be retrieve since it retrieved by the Time series. A considers the every time sequence which is recorded by the control instrument. To make retrieval of record for the generic repost provided by the physician. There are different modules through which the data flow &execution of project can understand. They are as given below;

1. ***Query Data Generation:-***

   Here the user needs to provide the inputreport for which the record has to retrieve on thebasis of Time series [2] [3]. The input report will notbe readable so it has to be converted intointermediate data which is readable by the program.Then the data present in the intermediate data will beretrieved and provided for the physician. Then thedata will be process for modifying the data in thereport if necessary. Then the finalized data will begenerated for the further processing.

2. Domain Selection:-

After acquiring the query data, the domain in which the query data have to be deal with will have to be analyzed [4]. The domain will lies in the query report by the user. Then the required record based on the domain will be clustered. Then the record for the respective domain will be retrieved.

3. Retrieving Record:-

Here the record that suits the query domain will be retrieved based on analyse of the input query data provided by the physician. Here the retrieved records will of same domain but different dimensionality that this record will be containing overall records for the domain provided in the query data. So it has to be gathered in the appropriate way[4].

4. Temporal Abstraction:-

The Outcome of retrieving records contains records of same dimensionality but different dimensions. So it's necessary to retrieve most accurate record for the query data

provided by the physician. It retrieval of the resultant record will carried out by the temporal abstraction. For implementing the Temporal Abstraction an Efficient the Viterbi algorithm is used for enhancing the temporal abstraction [1]. Temporal abstraction will analyse the each and every dimension of the records with respect to Time series [2] [8]. Then the records that match the query data will be retrieved. It will be shortlisted record of the required set from the overall domain record. Then the most accurate dataset will be retrieved from the featured records. Then the detailed information of the accurate record will be provided to the user.

5. Generation of Report:-

After acquiring the most accurate record that matches the query data, it has to provide to the user in the most precise form. Hence in this module the finalized information will be saved in magnetic disk and also user can able to take printout of the information.

III DEVELOPED BASIC CONSTRUCTS FOR MAINTAINING CLASSIFICATION STATISTICS

The moments in time at which the summary statistics arestored are organized in the form of a geometric time frame.The use of the geometric time frame provides the flexibilityof the classification model. This is because the microclusters,which are stored at different moments in time, can beused to quickly construct a classification model overdifferent time horizons. At each moment in time, theclassification on the test stream is performed by using ahorizon which suits the needs of that particular moment.While the work in [2] discusses a pyramidal time frame inthe context of the clustering problem, the geometric timeframe provides a more effective implementation of thestorage process. We note that the concept of a geometrictime frame is different from the pyramidal frame in terms ofreducing the overlap among different snapshots of the data.It is assumed that the training and test data streams eachconsist of a set of multidimensional records X1 . . .Xk. . .arriving at time stamps T1 . . . Tk. . . . Each Xi is a multidimensionalrecord containing d dimensions which aredenoted by Xi ¼ ðx1i. . . xdiÞ. In addition, each record Xi inthe training data stream is associated with a class label Cj.

We assume that the class_id of the class Cjis j.We will first begin by defining the concept of supervisedmicroclusters. While the microclustering concept of [2] isuseful for unsupervised clustering, we need to makemodifications in order to use this approach for theclassification process. The supervised microclusters arecreated from the training data stream only. Each suchmicrocluster corresponds to a set of points from the trainingdata, all of which belong to the same class.Definition 2.1.A supervised microcluster for a set ofd-dimensional points Xi1 . . .Xinwith time stamps Ti1 . . . Tinand belonging to the class class id is defined as the ð2 _ d þ 4Þtuple ðCF2x, CF1x, CF2t, CF1t, n, class idÞ, whereinCF2x and CF1x each correspond to a vector of d entries. Thedefinitions of each of these entries are as follows:

1.For each dimension, the sum of the squares of the datavalues are maintained in CF2x. Thus, CF2x containsd values. The pthentry of CF2x is equal toPnj¼1ðxpijÞ2.

2.For each dimension, the sum of the data values aremaintained in CF1x. Thus, CF1x contains d values.The pthentry of CF1x is equal to Pnj¼1 xpij.

3.The sum of the squares of the time stamps Ti1 . . . Tinare maintained in CF2t.

4.The sum of the time stamps Ti1 . . . Tin are maintainedin CF1t.

5.The number of data points are maintained in n.

6.The variable corresponding to class id corresponds tothe class label of that microcluster.

The above definition of the supervised microcluster forthe set of points C is denoted by CFTðCÞ. This summaryinformation is an extension of the cluster feature vectorconcept discussed in . Since each component in thedefinition of the microcluster is an additive sum overdifferent data points, this data structure can be updatedeasily over different data streams. We note that themicroclustering construct is primarily designed for the caseof continuously defined attributes. In order to handlecategorical data, a similar construct needs to be designedfor such variables; a task which is beyond the scope of thispaper.

Frame no.	Snapshots (by clock time)
0	69 67 65
1	70 66 62
2	68 60 52
3	56 40 24
4	48 16
5	64 32

TABLE 1: A Geometric Time Window

Frame no.	Snapshots (by clock time)
0	70 69 68
1	70 68 66
2	68 64 60
3	64 56 48
4	64 48 32
5	64 32

TABLE 2: A Pyramidal Time Window

IV EXPERIMENTAL RESULTS

4.1 Comparison with Pyramidal Time Frame

The process of maintenance of supervised microclustersbelonging to different classes derives ideas from the nearest-neighbor and k-means algorithms. Because of the supervised nature of the method, class labels need to be used during the clustering process. At any moment in time,

a maximum of q microclusters are maintained by the algorithm. We denote these microclusters by M1 . . .Mq. Associated with each microcluster i, we create a unique id whenever it is first created. As we shall subsequently see, the microcluster maintenance algorithm requires a merging of multiple microclusters into one microcluster. Only microclusters that belong to the same class may be merged together during the clustering process. When two such microclusters are merged, a list of ids is created in order to

identify the constituent microclusters. The value of q is determined by the amount of main memory available in order to store the microclusters. The microclusters whichare maintained in main memory correspond to the currentsnapshot of summary statistics.

1.Asmall portion of the stream is used for the processof horizon fitting. The corresponding portion of thetraining stream is referred to as the horizon fittingstream segment. The number of points in the dataused is denoted by kfit. We note that the value of kfitis typically very small, such as 1 percent of the data.

2. The remaining majority of the stream is used foraccumulation of the pertinent statistics correspondingto the microclusters and class information.

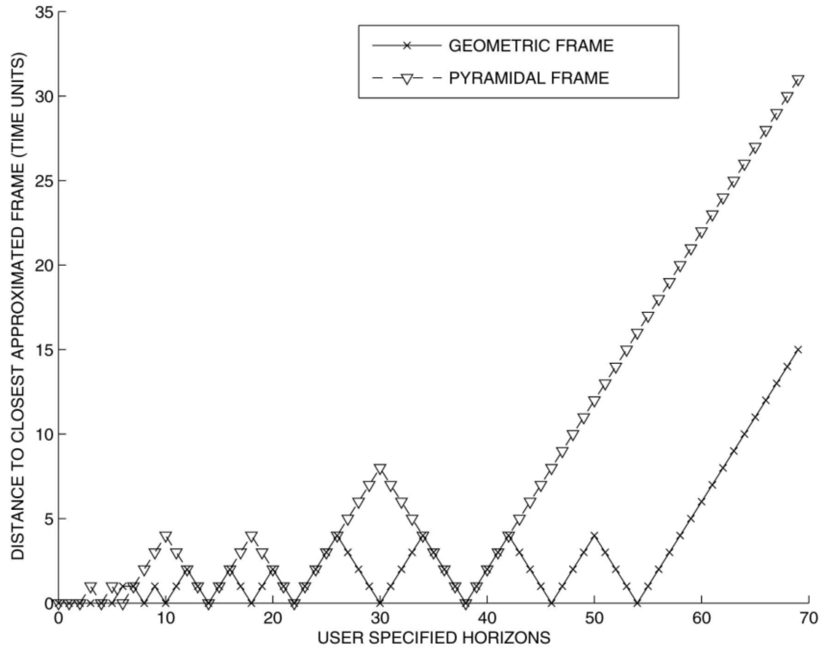

Fig. 1.Comparison between the pyramidal and geometric time window.

V CONCLUSION

In this devloped framework for the classification of dynamic evolving data streams by adapting a previously developed (unsupervised) approach for microclustering[2]. While previous research has developed methods on the development of one-pass algorithms for data stream classification, this paper proposes a new framework and a different methodology for online classification and continuous adaption to fast evolving data streams. The stream classification framework proposed in this
study has the following fundamental differences from the previous stream classification work in design philosophy.
First, due to the dynamic nature of evolving data streams, Second, stream data classification needs to use temporal data and historical summary in its analysis. However, it is too costly to keep track of the entire history of data in uniform, fine granularity. Thus, a geometric time window, with more recent time in finer granularity and more remote time in coarser granularity, strikes a good balance. Previous studies on stream data clustering and stream time-series analysis present logarithmic window [4], natural pyramidal time window [7], and pyramidal time window [2]. This documentdesign of geometric time window follows this trail but strikes a balance between logarithmic compression and sufficiently detailed coverage of recent events. Geometric time window gives us great flexibility at selection of the appropriate horizon in stream classification. It also provides good potential to carry out other powerful stream classificationtasks, such as construction and comparison of models at different time frames, discovery of the evolutionof models with time, and so on. This should be an interesting issue for further study.Third, for dynamic model construction, compression should be performed not only along with time, but also onthe data objects themselves due to the numerocity of the data. In order to achieve this goal, summary data is generated using microclustering in conjunction with a geometric time window. classification inboth continuous query mode (as a built-in watchdog) andad hoc query mode upon user's mining request. In summary, we have proposed an interesting framework for online classification of dynamically evolving data streams. The new framework has been designed carefully based on our analysis and reasoning and has been tested based on our experiments on a real intrusion detection data set.

ACKNOWLDGEMENT

This is great opportunity to acknowledge and to thanks everyone without their support and help this paper would have been impossible. Firstly I would like to thank my guide, Prof.P.D.Sathya, for his guidance and support. I will forever remain grateful for the constant support and guidance extended by guide, in making this paper. Through our many discussions, he helped me to form and solidifyideas. I would like to extend my special thanks to my Family and Abhijeet Barde for their moral support and valuable suggestions.

REFERENCES:

[1] C.C. Aggarwal, J. Han, J. Wang, and P. Yu, "On Demand Classification of Data Streamsm," Proc. ACM KDD Int'l Conf.Knowledge Discovery and Data Mining, pp. 503-508, Aug. 2004.

[2] C.C. Aggarwal, J. Han, J. Wang, and P. Yu, "CluStream: A Framework for Clustering Evolving Data Streams," Proc. Int'l Conf. Very Large Data Bases, pp. 81-92, Sept. 2003.

[3] C.C. Aggarwal, "A Framework for Diagnosing Changes in Evolving Data Streams," Proc. ACM SIGMOD Conf., pp. 575-586, June 2003.

[4] B. Babcock, S. Babu, M. Datar, R. Motwani, and J. Widom, "Models and Issues in Data Stream Systems," Proc. 21st ACM SIGACT-SIGMOD-SIGART Symp. Principles of Database Systems, pp. 1-16, June 2002.

[5] L. O'Callaghan, N. Mishra, A. Meyerson, S. Guha, and R. Motwani, "Streaming-Data Algorithms For High-Quality Clustering," Proc. 18th Int'l Conf. Data Eng., pp. 685-696, Feb. 2002.

[6] P. Bradley, U. Fayyad, and C. Reina, "Scaling Clustering Algorithms to Large Databases," Proc. Knowledge Discovery and Data Mining Conf., pp. 9-15, 1998.

[7] Y. Chen, G. Dong, J. Han, B.W. Wah, and J. Wang, "Multi-Dimensional Regression Analysis of Time-Series Data Streams," Proc. 28th Int'l Conf. Very Large Data Bases, pp. 323-334, Aug. 2002.

[8] P. Domingos and G. Hulten, "Mining High-Speed Data Streams," Proc. Sixth ACM SIGKDD Int'l Conf. Knowledge Discovery and Data Mining, pp. 71-80, Aug. 2000.

[9] P. Domingos and G. Hulten, "A General Method for Scaling Up Machine Learning Algorithms and Its Application to Clustering," Proc. Int'l Conf. Machine Learning, pp. 106-113, 2001.

[10] R. Duda and P. Hart, Pattern Classification and Scene Analysis.New York: Wiley, 1973.

[11] J.H. Friedman, "A Recursive Partitioning Decision Rule for Non- Parametric Classifiers," IEEE Trans. Computers, vol. 26, pp. 404-408, 1977.

[12] J. Gehrke, V. Ganti, R. Ramakrishnan, and W.-Y.Loh, "BOAT: Optimistic Decision Tree Construction," Proc. 1999 ACM SIGMOD Int'l Conf. Management of Data, pp. 169-180, June 1999