An Adjusted Methods on Classification Algorithm for Streaming Data

Hind Ra'ad Ibraheem[1] Enas Mohammed Hussein [2]

Research Scholar [1 2], Professor[3]

Department of Computer Science

Al mustansiriya university college education

Baghdad

Iraq

Abstract

In recent years, advances in hardware technology have facilitated the abilityto collect data continuously. Simple transactions of everyday life such as using a credit card, a phone or browsing the web lead to automated data storage. Similarly, advances in information technology have led to large flows of data across IP networks. In many cases, these large volumes of data can be mined for interesting and relevant information in a wide variety of applications. When the volume of the underlying data is very large, it leads to a number of computationaland mining challenges. Streaming data is potentially endless of incoming data at high speed and may evolve over time.the data stream has recently emerged in response to the continuous data problem. The algorithm processing the stream has no control over the order of the examples seen, and must update its model incrementally as each example is inspected. Performance of data stream classification is measuring by involving processing speed, memory and accuracy. Also A classification algorithm must meet several requirements in order to work with the assumptions and be suitable for learning from data streams that is process an example at a time and inspect it only once; use limited amount of memory.Similar to data mining, data stream mining includes classification, clustering, frequent pattern mining etc. techniques; the special focus of this paper is on classification methods invented to handle data streams.This paper discuss two improves manners on Hoeffding tree algorithm a well-known classification data stream algorithm. Both improves is based on tie breaking parameter. The first improve named Modify Hoeffding Tree Algorithm (MHTA) and the second one named Variable Random Tie Generating Values Algorithm (VRTGVA) .

Keywords: *Stream data classification, Hoeffding tree, MHTA,VRTGVA.*

1. INTRODUCTION

Data streams have received a lot of attention over the last decade, which is an important aspect in real-world applications like Credit card operations, sensor networking and banking services.Database transactions, telecommunication services generate logs and other forms of stream data. The generated data by these applications is dynamic which is difficult to handle and organize[1]. Data stream mining algorithms extract information from volatile streaming data. Stream data algorithm sometimes cannot process the data more than once. So, the algorithms have to be designed such that they work effectively in that single pass only. Stream data classification has limited power and

memory, which cannot handle and store gigantic volume of traffic as well [2]. For the last few years, most of the applications have been working on stream data, widely used in Peer to Peer a (P2P) application which includes Bit Torrent, Emule, Kaaza etc., resulting in increased internet traffic. These applications increase the internet traffic by around 85% and create huge amounts of internet data. Several messenger-based applications like Yahoo and Google Talk, used by most people in peak hours, are again a major reason to rise in internet traffic. Some other most-used applications like web, e-mails and file transfer also increase the internet traffic data significantly. Traditional data mining algorithms work on the assumption that they will have sufficient resources to process particular data. This assumption does not have any chance in data stream mining due to continuous evolvement of new data. Every Stream data mining algorithms should take less time to learn provided data with few amount of memory [3].Classification in data stream has some challenges that researchers attempt to solve them. Three main challenges of classification techniques are as follows:

- Accuracy: It is the most important factor in classification algorithms, and concept drifting directly influences the accuracy.
- Efficiency: creating of a classifier is costly from processing point of view. Also, updating of the model is a challenge due to drifting.
- Ease of use: a classifier model should be usable in applications[4].

2. MINING TASK

Stream mining task includes task like Classification, Clustering and Mining Time-Series Data. In this paper, we will discuss a two adjusted methods on Hoeffding tree algorithm based on tie breaking parameter. That are used for classification of stream data and their comparison based on their experimental results.

Classification generally is a two-step process consisting of learning or Model Construction (where a model is constructed based on class labeled tuples from training set) and classification or Model Usage (where the model is used to predict the class labels of tuples from new data sets).

3. RELATED WORK

- One of the pioneer works in decision tree induction for the streaming setting is the Very Fast Decision Tree algorithm (VFDT) [5]. This work focuses on alleviating thebottleneck of machine learning application in terms of time and memory, i.e. the conventional algorithm is not able to process it due to limited processing time and memory. Its main contribution is the usage of the Hoeffding Bound to decide the number of data required to achieve certain level of confidence. This work has been the basis for a large number of improvements, such as dealing with concept drift and handling continuous numeric attributes.
- Ben-Haim and Tom-Tov [6] present an algorithm for building decision trees in a streaming and parallel setting. The distribution of attributes from the master process proceeds in a horizontal fashion, and the tree is built while taking advantage of histograms of the data maintained at the working processes. However, the horizontal splitting can quickly increase the memory overhead of the replicated model and challenge its scalability.

- KourtellisN. et al, in [7] present the Vertical Hoeffding Tree (VHT), the firstdistributedstreaming algorithm for learning decision trees. It features anovel way of distributing decision trees via vertical parallelism.The algorithm is implemented on top of Apache SAMOA, aplatform for mining big data streams, and thus able to run onreal-world clusters. Our experiments to study the accuracy andthroughput of VHT, prove its ability to scale while attaining
  superior performance compared to sequential decision trees.
- *Yael Ben-H. and Elad Tom-T.,* presented [8] "A Streaming Parallel Decision Tree Algorithm". It gave a new algorithm for building decision tree classifiers for classifying both large data sets and streaming data. The essence of the algorithm is to quickly construct histograms at the processors, which compresses the data to a fixed amount of memory because of the large number of training examples. It is not feasible to store the examples (even in each separate processor). Therefore, a processor can both save a short buffer of examples and uses them to improve (or construct) the classifier, or builds a representative summary statistic from the examples.

4. The suggested manners

As mentioned before, our suggested methods is an improvement on Hoeffding tree algorithm. The suggested methods is based on tie breaking parameter which has an effect on splitting process, it means converting a internal node into terminal node (the node contains class label). When two candidates of nodes competing to become a splitting node are equally good (having almost the same value of information gain), it may take a long time and intensive computation to decide between them. This situation not only drains significant amounts of computational resources, but the tie-breaking result at the end might not always contribute substantially to the overall accuracy of the decision tree model.Traditional Hoeffding tree algorithm set tie *t*into 0.05 as a default value. In our proposal, both proposed methods are based on *t* in different way as will illustrated below.

First we will make a review to the Hoeffding algorithm, The Hoeffding tree (a.k.a.VFDT) is a streaming decision tree learner with statistical guarantees. In particular, by leveraging the Chernoff-Hoeffding bound , it guarantees that the learned model is asymptotically close to the model learned by the batch greedy heuristic, under mild assumptions. The learning algorithm is very simple. Each leaf keeps track of the statistics for the portion of the stream it is reached by, and computes the best two attributes according to the splitting criterion.A Hoeffding tree is capable of learning from massive data streams with assumption that the distribution generating examples do not change over time. Classification problem is a set of training examples ofthe form (m, n), where ‘m’ is a vector of n attributes and n is a discrete class label. The objective is to produce a model n=f (m) so as to provide and predict the classes n for future examples m with high accuracy. Decision tree learning is a powerful technique in classification. Decision tree learning node has a check on attributes and each branch providing output of the check[9].

4.1A General View of the Suggested Methods

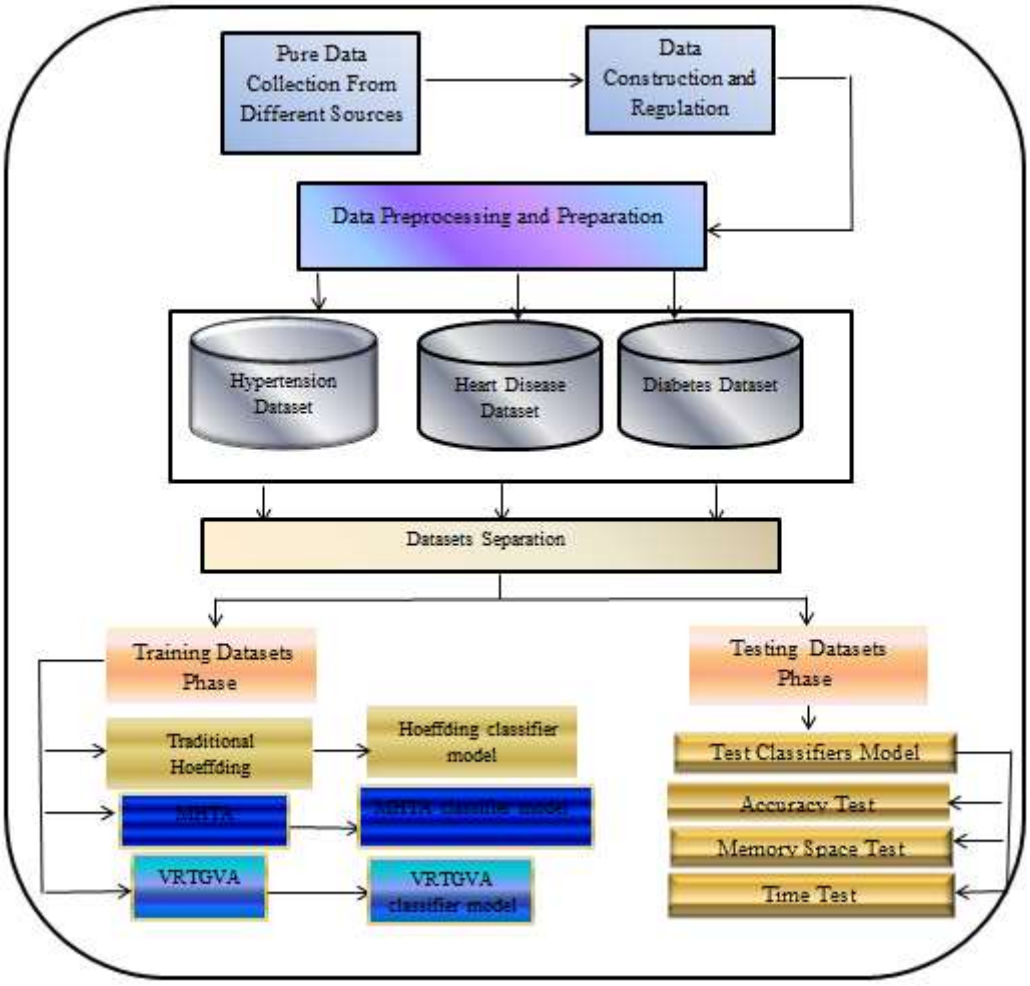

Figure 1 Diagram of Suggested Manners

4.1.1 Pure Data Collection from Different Sources

At this stage, pure data were gathered from heterogeneous sources. Three various medical data were obtained, the first data was hypertension disease gathered and information extracted manually from patients files from Iraqi hospital (real world). The second data are heart disease and the third one are diabetes disease were taken from the internet.

4.1.2 Data Construction and Regulation

Here these collections of diversedatawere constructed analyzed. Each data disease was organized according to the set of disease indicators and its possible values.

4.1.3 Data Preprocessing and Preparation

It is ever the most significant step. Today's actual-world data bases are highly over sensitive to noisy, missing, and inconsistent data due to their typically vast size and their likely origin from manifold, heterogeneous sources. The preprocessing assist to enhance the quality of the data and, consequently, of the mining results and to make the knowledge discovery extra efficient to improve the efficiency and ease of the mining process. Many operations are

performed on the original data manually, in order to prepare and make the data more convenient to be used for stream mining process.

4.1.4 Datasets Separation

After the datasets had been processed, a simple splitting partitions are conducted to the datasets that divided it into two subsets which are the training set (which the algorithm is applied on it to build the classifiers models) and the second one are the testing subset (these subsets used to evaluated the induced classifier).It is common to designate (2/3) of the data as training data sets and (1/3) of the data as testing data sets. This stage includes two phases:-

- *Training phase*: involves building a classifiers models induced by applying above mentioned four classification algorithms.
- *Testing phase*: the evaluation of the induced classifiers in this step.

4.1.5 Modify Hoeffding Tree Algorithm (MHTA)

MHTA is the first proposed algorithm based on the tie breaking threshold. The algorithm be carrying out in all available three training datasets in order to induced a classifier model. The suggested method is a modern devise version of the original Hoeffding tree algorithm which uses an adjusted tie with many generating values, that can provide good classification accuracy and regulate the growth of decision tree size to a reasonable extent**.** As explained inthe algorithm below:

Algorithm (3-1): Modify Hoeffding Tree Algorithm (MHTA).
Input: N*min*, $^{\delta}$, N*l* = 0,*t* =0.0 ,inc-value, max-inc and a set of training examples. **Output**: Decision tree, classification accuracy, memory space and execution time.

Process

1. Start a tree with a single leaf node *f* (the root).
2. **For** all training examples **do**

Begin

2.2.1 Using Hoeffding tree(*HT*) to sort the examples.
2.2.2 Modify the sufficient statistics in *f*
2.2.3 Increment N*l*.
2.2.4 **If** N*l* **mod** N*min*=0 **and** examples seen not regard to the same class **then**
Begin
a. Calculate the gain G for each attribute (Yi).
b. Pick out the two highest gain values as Ya and Yb.
c. Let Y0 = the attribute with less gain value(NULL).
d. Compute R = Ya- Yb.
e. Calculate HB (€) using equation (2.9).
f. t=t+inc_value
g. if t ≠ max_inc then
begin
If Ya<> Y0 and (R > € or € <*t*)
then
Displace *f* with an interior node that incises on Ya.
Else return to step f.
For all branches of the incise do
Create a new leaf with initialized sufficient statistics.
End For
Compute classification accuracy, execution time and memory space
Else return to step f
End IF
Else return to step 2.2.3
End IF
2.2.5 Return *t*-value with the highest accuracy, execution time & memory space

End For
End Process

4.1.6 Variable Random Tie Generating Values Algorithm(VRTGVA)

The second suggested method will called variable random tie generating values (VRTGV). This algorithm is depending on the tie breaking parameter as well, but here instead of taking a sequential values in a specific range, a set of M random values will take according to the size of datasets. In another word there are many different sizes of the three medical datasets samples had been used, on each dataset size different M of *t* values were used. For example, on dataset which have 10000 example its M values were 25,by took M values in random fashion (which means diverse random M values). As shown in algorithm below:

Algorithm (3.2): Variable Random Tie Generating Values (VRTGV).

Input : Nmin, max_iteration, $^{\delta}$ = 0.0001, N*l* = 0, *t* = 0.0 , iteration=0 and a set of training examples.
Output: Decision Tree , Classification Accuracy , Execution Time and Memory Space.

Process:
Begin
1. Start a tree with a single leaf node *f* (the root).
2. **For** all training examples **do**

Begin
2.2.1 using Hoeffding tree(*HT*) to sort the examples
2.2.2 modify the sufficient statistics in *f*
2.2.3 increment N*l*
2.2.4 **IF** N*l* **mod** Nmin = 0 **and** examples not belong to the same class **then**

Begin
a. Calculate the Gain G for each attribute (Yi)
b. Pick out the two highest attribute gain as Ya and Yb
c. Let Y0 = the attribute with less gain value(NULL)
d. Compute R = Ya - Yb
e. Calculate HB (€) using equation (2.9)
f. iteration=iteration +1
g. if iteration ≠ max_iteration then

begin
- ***t = create random value( )***
- **IF** Ya<>Yb**and**(R>€ **or** €<*t*) **then**

- Displace *f* with an interior node that incise on Ya

Else **return to step f**
- **For** all branches of the incise **do**
  - Create a new leaf with initialized sufficient statistics

End For
- Compute classification accuracy, execution time and memory space
- ***Until t= max value***

Else return to return to step f

End IF
Else return to step 2.2.3
End IF
2.2.5 Return *t* value with highest accuracy , execution time and memory space
End for
End Process

5. EXPERIMENTS AND RESULTS

Different datasets sizes were used. It was 10000, 25000, 50000, 100000 for the three datasets. After applying the traditional Hoeffding algorithm, MHTA and VRTGVA, the results below were obtained according to the measurements were used which are, classification accuracy, memory space and execution time.

Table 1Hoeffding vs. MHTA & VRTGVA for Hypertension Dataset.

Data set Size	Training records number	Testing records number	M Values	Hoeffding Accuracy (%)	MHTAAccuracy (%)	VRTGVA Accuracy (%)
10000	6667	3333	25	47.7047	72.7668	70.0067
25000	16667	8333	50	48.3739	71.6260	76.4456
50000	33333	16667	75	49.1690	73.5929	71.5509
100000	66667	33333	100	50.2234	72.8443	75.2236

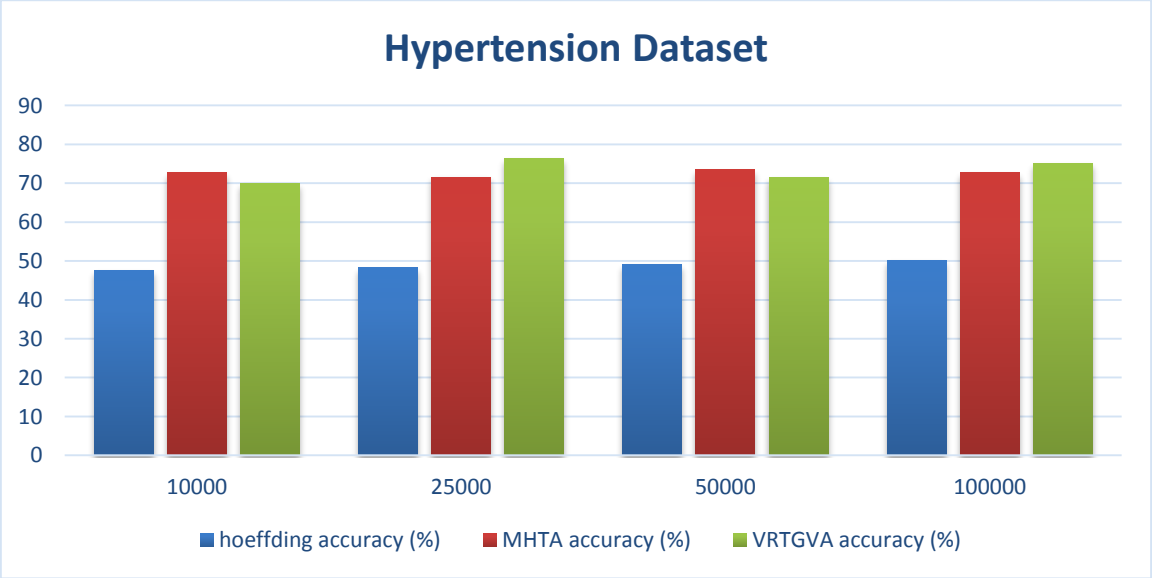

Figure 2 Hypertension DatasetChart for Hoeffding vs. MHTA& VRTGVA

As shown in table 1 and figure 2, MHTA and VRTGVA were obtained highest accuracy than traditional Hoeffding algorithm.

Table 2 Hoeffding vs. MHTA& VRTGVA for Heart Disease Dataset.

Data set size	Training records number	Testing records number	Hoeffding execution time (millisecond)	MHTAExecution time (millisecond)	VRTGVA Execution time (millisecond)
10000	6667	3333	42	16	14
25000	16667	8333	97	36	40
50000	33333	16667	105	98	101
100000	66667	33333	243	220	218

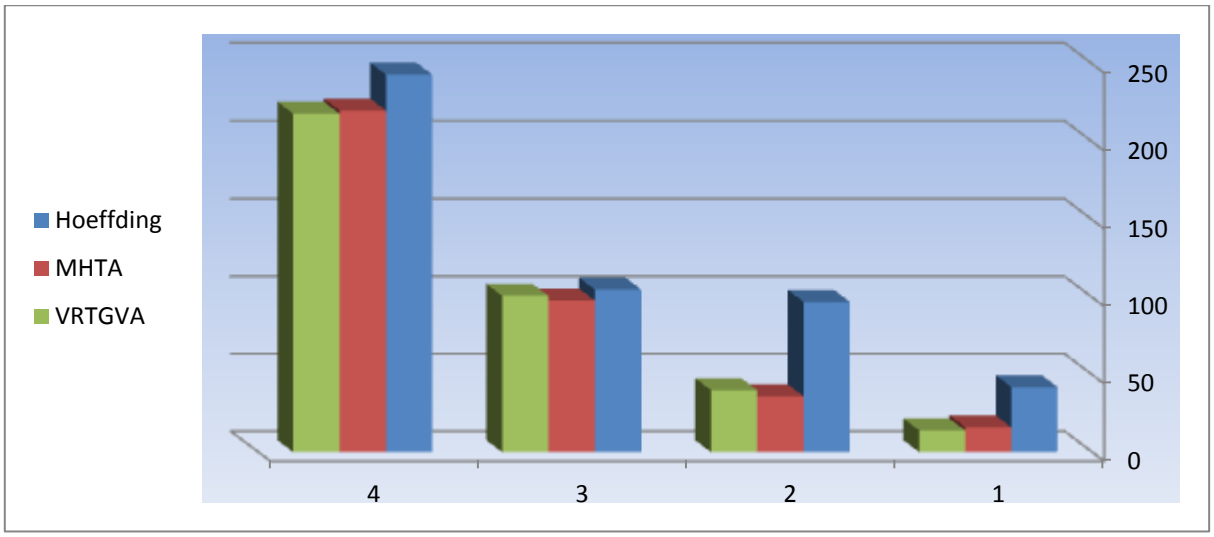

Figure 3 Heart Disease DatasetChart for Hoeffding vs. MHTA& VRTGVA

As shown from table 2, figure 3 above, MHTA and VRTGVA were obtained lesser execution time comparing with the traditional Hoeffding algorithm.

Table 3 Hoeffding vs. MHTA & VRTGVA forDiabetes Dataset.

Data set size	Training records number	Testing records number	Hoeffding Memory space (byte)	MHTAMemory space (byte)	VRTGVA Memory space (byte)
10000	6667	3333	3784	9066	3500
25000	16667	8333	13399	5773	5780
50000	33333	16667	14699	39471	13600
100000	66667	33333	34567	3784	3784

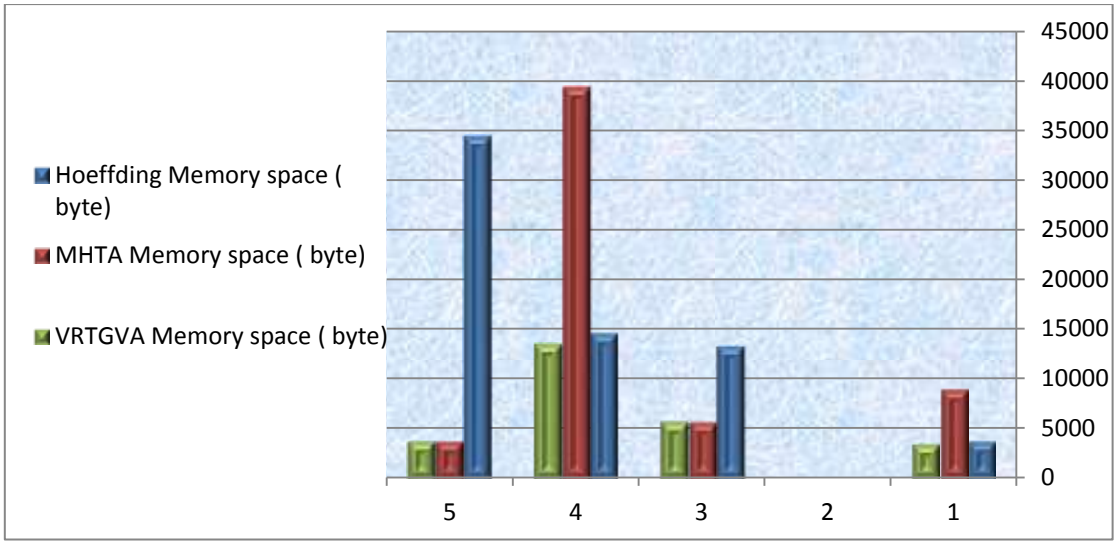

Figure 4. Diabetes Dataset Chart for Hoeffding vs. MHTA& VRTGVA.

As shown from table 3, figure 4, the results were diverse, traditional Hoeffding algorithm and VRTGVA were obtained lesser memory space in contrast with the MHTA on two datasets sizes (10000 and 50000).

6. CONCLUSION

A two modified classification techniques were introduced to assure on creating new advantages and eliminating the previous disadvantages for the other classification techniques. Thus, it could be concluded that the proposed adjusted methods has achieved the following:

a. MHTA and VRTGVA has achieved a higher accuracy comparing with the traditional Hoeffding algorithm. This means that the mining process will achieve more accurate results comparing with traditional Hoeffding algorithm.

b. MHTA and VRTGVA has achieved a lower execution time comparing with the traditional Hoeffding algorithm. This means that the mining process will achieve more processing speed comparing with traditional Hoeffding algorithm.

c. MHTA and VRTGVA has achieved a lower need to memory space comparing with the traditional Hoeffding algorithm. This means that the mining process will need less memory storage which leads to accomplish a lower cost technique comparing with traditional Hoeffding algorithm.

REFERENCES

1. HomayounS. andAhmadzadehM., "A review on data stream classification approaches", Department of Computer Engineering and Information Technology, Shiraz University of Technology, Shiraz, Iran,Journal of Advanced Computer Science & Technology, 2016.
2. Kourtellis N. et al, "VHT: Vertical Hoeffding Tree", Telefonica I+D, Spain, Telecom ParisTech, France, 2015.

3. Ms. Madhu S. S. and Ms. Madhu S. S., " Stream Data Mining and Comparative Study of Classification Algorithms ", Department of Computer Engineering), (C.U.Shah College of Engineering and Technology, Gujarat, India ,International Journal of Engineering Research and Applications (IJERA) , Vol. 3, Issue 1, January -February 2013.

4. G. De Francisci Morales. SAMOA:" A Platform for MiningBig Data Streams", 2013.

5. Yang H. and Fong S., "Moderated VFDT in Stream Mining Using Adaptive TieThreshold and Incremental Pruning" , Department of Science and Technology, University of Macau, 2013.

6. Hulten, G., Spencer, L., Domingos, P.: "Mining time-changing data streams",In: Proceedingsof the Seventh ACM SIGKDD International Conference on Knowledge Discovery and DataMining. KDD 2001, pp. 97–106. ACM, New York ,2001.

7. Arvind K., Parminder K.and Pratibha S.," A Survey on Hoeffding Tree Stream Data Classification Algorithms" , 3Department of Computer Science and Engineering,National Institute of Technology, Hamirpur-177005, India, CPUH-Research Journal: 2015.

8. Hulten, G., Spencer, L., and Domingos, P. "Mining time-changing data streams" , In Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining 97-106, 2001

9. J. Gama, R. Rocha, and MedasP., " Accurate decision treesfor mining high-speed data streams". In SIGKDD, 2003.