International Journal of Engineering and Advanced Technology (IJEAT)
ISSN: 2249–8958, Volume-1 Issue-3, February 2012

Comprehensive study on techniques of Incremental learning with decision trees for streamed data

Prerana Gupta, Amit Thakkar, Amit Ganatra

Abstract— Incremental learning is an approach to deal with the classification task when datasets are too large or when new examples can arrive at any time. Data streams are inherently time-varying and exhibit various types of dynamics. There are some problems in data stream mining like class imbalance, concept drift, arrival of a novel class, etc. This paper focuses on the problem of concept drift. The presence of concept drift in the data significantly influences the accuracy of the learner, thus efficient handling of non-stationary environment is an important problem. Detecting changes of concept definitions in data streams and adapting classifiers to them is studied in this paper. The classifying technique studied is decision trees classification for streamed data, As decision trees are more efficient and easily interpretable. The comparative studies of some algorithms FIMT-DD, ORTO, FIOT, OVA-classifier, i+learning, UFFT, SCRIPT and HOT are shown in this paper

Index Terms— concept drift, Data stream mining, Incremental learning, Hoeffding trees

I. Introduction

Data mining, an important technique used in searching for knowledge in databases, has attracted many researchers' attention in recent years. Among several functionalities of data mining, classification is crucially important and has been applied successfully to several areas (Jegeleviˇcius et al., 2002; Remeikis et al., 2004). The popular techniques developed for classification includes Bayesian classification, Neural Networks, Genetic Algorithms, and Decision Trees (Gehrke et al., 2000; Han and Kamber, 2001; Mehta et al., 1996; Miseviˇcius, 2006; Quinlan, 1993. Classification is one of the main tasks in machine learning and data mining. Many researchers have been undertaken and a great number of methods, based on different principles, have already been proposed [1]. However, in addition to displaying comparable classification accuracy to other techniques, decision tree is more efficient and easily interpreted by human (Rastogi and Shim, 1998). In the research domain of decision tree, several important issues, including scalability (Gehrke et al., 2000), imbalanced dataset (Japkowicz and Stephen, 2002), ensemble classifier (Chawla et al., 2002), incremental learning (Schlimmer and Fisher, 1986; Utgoff, 1989) etc., also have been widely studied. Since the real-world data nowadays might come in the form of consecutive data blocks (Domingos and Hulten, 2000), researchers have put more and more attention on data streams mining. The related applications include e-mail sorting (Cohen, 1996), calendar scheduling (Blum, 1997), and computer intrusion detection (Maloof and Michalski, 2000) etc. Nevertheless, most proposed approaches of data stream mining assumed data blocks come under stationary distribution.

Incremental learning is an approach to deal with the classification task when datasets are too large or when new examples can arrive at any time [2]. An algorithm possesses incremental learning capabilities, if it meets the following criteria [10]:

1. Ability to acquire additional knowledge when new datasets are introduced.
2. Ability to retain previously learned information.
3. Ability to learn new classes if introduced by new data.

ID3 is a useful concept learning algorithm because it can efficiently construct a decision tree that generalizes well. For non incremental learning tasks, this algorithm is often a good choice for building a classification rule. However, for incremental learning tasks, it would be far preferable to accept instances incrementally, without needing to build a new decision tree each time. ID4 algorithm was designed to learn decision trees incrementally, and then presents a new incremental tree construction algorithm, named ID5R. The ID5R algorithm builds the same tree as the basic ID3 tree construction algorithm, given the same instances and given the same method for breaking ties among equally good attributes. Restructuring of the decision tree takes place recursively so that the desired test attribute is at the root [9]. Other decision

Manuscript published on 28 February 2012.
* Correspondence Author (s)
Prerana Gupta, Department of Computer Engineering, Charotar Institute of Technology (Faculty of Technology and Engineering), Charotar University of Technology Changa, Anand, Gujarat, India, 388421 prerana_gupta2006@yahoo.co.in
Amit Thakkar, Department of Information Technology, Charotar Institute of Technology (Faculty of Technology and Engineering), Charotar University of Technology Changa, Anand, Gujarat, India, 388421 amitthakkar.it@ecchanga.ac.in
Amit Ganatra, Department of Computer Engineering, Charotar Institute of Technology (Faculty of Technology and Engineering), Charotar University of Technology Changa, Anand, Gujarat, India, 388421 amitganatra.ce@ecchanga.ac.in

tree learning algorithms are C4.5, CART, CHAID, SPRINT, SLIQ and some more are there.

A data stream is an ordered sequence of instances that arrive at a rate that does not permit to permanently store them in memory. Data streams are potentially unbounded in size making them impossible to process by most data mining approaches. The main characteristics of the data stream model imply the following constraints [11]:

1. It is impossible to store all the data from the data stream. Only small summaries of data streams can be computed and stored, and the rest of the information is thrown away.

2. The arrival speed of data stream tuples forces each particular element to be processed essentially in real time, and then discarded.

3. The distribution generating the items can change over time. Thus, data from the past may become irrelevant or even harmful for the current summary.

In almost all real-life situations, the concepts we are trying to model may change during the course of learning, especially when data are analyzed over longer periods of time. This is an important issue when learning from data streams due to their dynamical nature. The changes in the hidden variables or context induce changes in the target concept which is known as concept drift. Different techniques are studied. Techniques have their algorithms which are evaluated extensively in a variety of settings involving artificial and real data. To the best of our knowledge there is no other general purpose algorithm for incremental learning regression/model trees able to perform explicit change detection and informed adaptation. The algorithms perform online and in real-time, observes each example only once at the speed of arrival, and maintains at any-time a ready-to-use model tree. The tree leaves contain linear models induced online from the examples assigned to them, a process with low complexity. The algorithms have mechanisms for drift detection and model adaptation, which enable it to maintain accurate and updated regression models at any time. The drift detection mechanism exploits the structure of the tree in the process of local change detection. As a response to local drift, the algorithm is able to update the tree structure only locally [4].

II. THE PROBLEM OF CONCEPT DRIFT IN DATA STREAM MINING

To make readers easily understand the problem we will address later, in this paper we divide the concept drift into concept stable, concept drift, and concept shift (Hsieh, 2004). We refer to the examples in (Wang et al., 2003) and modify the figures to illustrate the problem in Fig. 1. Fig. 1 represents a two-dimensional data stream and is divided into six continuous data blocks according to the arriving time of data. Instances arriving between ti and ti+1 form block Bi, and the separating line in each block stands for the optimum classification boundary in this block. During time t0 to t1, data blocks B0 and B1 have similar data distribution. That is, data stream during this period is stable. Thereafter in B2, some instances shows concept drift and the optimum boundary changes. This is defined as "concept drift". Finally, data blocks B4 and B5 have opposite sample distribution and this is defined as "concept shift". Obviously, since the sample distributions of the first two blocks B0 and B1 are quite close, we can use decision tree DT0 built by B0 as the classifier for B1 to save the computational and recording cost.

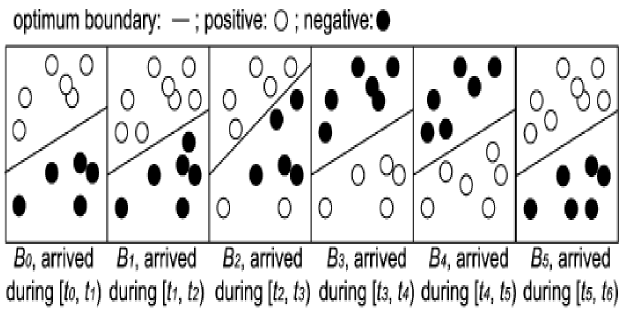

Fig.1. A data stream with the occurrence of concept drift [6]

Meanwhile, B2 shows slight differences when compared with the sample distribution of B1 and an efficient approach should make correction according to the original decision tree instead of rebuilding it [6]. We subdivide concept drift into one-way drift and two-way drift (Hsieh, 2004). Take Fig. 1 as the example again, we can find that some negative data in B2 drift to be positive data in B3, known as one-way drift. However, the positive data in B4 drift to be negative in B5, and vice versa, known as two-way drift. We can regard two-way drift as a kind of "local" concept shift if it occurs in the internal or leaf node of a decision tree. If the variation of information gain or gini index is used as the criterion to judge the occurrence of concept drift, e.g., the difference of information gain adopted in CVFDT, we can detect only one-way drift since the information gain obtained from B4 would the same as B5. It is worth to note that for the real data, two-way drift might happen. For example, a hacker in turn uses two computers with IP address x and y to send attack packages. When an internal node, which is learned from the first data block, splits the packages form x as safe and that from y as attack, there might be a contrary result learned from another data block. A similar condition might be found in trash mail protection, image comparison and so on.

III. RELATED WORK

In [4] Authors have developed an incremental algorithm for learning model trees to address these issues, named fast incremental model trees with drift detection (FIMT-DD). The algorithm starts with an empty leaf and reads examples in the order of arrival. Each example is traversed to a leaf where the necessary

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

statistics are updated. Given the first portion of instances, 1 the algorithm finds the best split for each attribute, and then ranks the attributes according to some evaluation measure. If the splitting criterion is satisfied it makes a spit on the best attribute, creating two new leafs, one for each branch of the split. Upon arrival of new instances to a recently created split, they are passed down along the branches corresponding to the outcome of the test in the split for their values. Change detection tests are updated with every example from the stream. If a change is detected, an adaptation of the tree structure will be performed. In [3] paper, authors have proposed incremental option trees for regression on fast nonstationary data streams (FIOT). The option nodes are introduced in order to improve the bias management in incremental learners, introduce ambiguous splits for learning under gradual concept drift and enable faster growth without instability in the splitting decisions. They have shown that the option tree is able to achieve better accuracy faster than a regular regression tree. This is especially pronounced for the data with gradual concept drift. Option nodes act as an improved look-ahead strategy and present an interpretable version of ensemble methods. In the [14] work presents an incremental learning algorithm appropriate for processing high-speed numerical data streams. The main contributions of this work are the ability to use multivariate splitting tests, and the ability to adapt the decision model to concept drift. While the former has impact in the performance of the system, the latter extends the range of applications to dynamic environments. To detect concept drift, we maintain, at each inner node, a naive-Bayes classifier trained with the examples that cross the node. While the distribution of the examples is stationary, the online error of naive- Bayes will decrease. When the distribution changes, the naive-Bayes online error will increase. In that case the test installed at this node is no more appropriate for the actual distribution of the examples. When this occurs the entire sub tree rooted at this node will be pruned.-UFFT. In this [12] paper, a practical useful tool – i+DiaKAW (Intelligent and Interactive Knowledge Acquisition Workbench) is proposed to automatically extract useful knowledge from massive medical data by applying various data mining techniques for supporting real medical diagnosis. To fulfill this effort, i+DiaKAW has been developed to provide a novel dynamic on-line decision tree learning scheme – i+Learning (intelligent and incremental learning) methodology, which makes up for the traditional incremental decision tree learning algorithms by concerning the new available features in addition to the new incoming instances. This [13] paper proposes to use the OVA (one versus all) classification scheme to classify streaming data. Different from conventional ensembles of multiclass classifiers, OVA comprises an ensemble of binary classifiers each specialized in solving a class of problems. They show the studies of OVA's theoretical complexity, advantages, challenges and solutions for data stream classification. It suggests the following. First, OVA's component classifiers are relatively uncorrelated in error. Thus, the ensemble's overall classification accuracy can be improved because its components have higher diversity. Second, OVA can react fast to concept changes. Upon receiving each labeled instance I, OVA only needs to revise two component classifiers, the one I truly belongs to and the one most likely to misclassify I. Third, feeding each labeled instance to only two classifiers is also a clever undersampling scheme, which can relieve the imbalanced training data problem of OVA learning. It can also increase OVA's efficiency of training and updating. In [16] Hulten et al. (2001) proposed to choose the best split attribute when building decision tree for data streams based on Hoeffding bound (Hoeffding 1963). After n independent observation of a real-valued random variable r with range R, the Hoeffding bound ensures that, with confidence 1-δ, the true mean of r is at least $\bar{r}$ -$\in$, where $\bar{r}$ is the observed mean of the samples and

$\epsilon = \left(\frac{\mathrm{R}^2 \ln\left(\frac{1}{\delta}\right)}{2nl}\right)$^1/2. Let G(Xi)

be the evaluation function used to choose test attributes. Assume G is to be maximized, and let Xa be the attribute with highest observed G after seeing n examples and Xb be the second-best attribute. Let $\Delta\bar{G} = \bar{G}$ (Xa) - $\bar{G}$ (Xb). Given a desired δ, the Hoeffding bound guarantees that Xa is the correct choice with probability 1-δ, if n examples have been seen at this node and $\Delta G > \in$. Then the node can be split using the current best attribute, and succeeding examples will be passed to the new leaves. a sample st exists at node N with probability wt in uncertain decision tree. For uncertain data, the count for n observation at node N is represented by expected count, which is defined as follows:

$$\mathrm{n} = \sum_{t=1}^{|S|} wt = PC(S) \quad (1)$$

Here S denotes fractional samples observed at node N, so we have

$$\epsilon = \left(\frac{\mathrm{R}^2 \ln\left(\frac{1}{\delta}\right)}{2PC(S)}\right)\text{^}1/2. \quad (2)$$

We name Hoeffding bound following formula (2) as probabilistic Hoeffding bound. Here, based on probabilistic Hoeffding bound, we use the uncertain information gain as the split evaluation function to choose the best splitting attribute. We extend CVFDT to our UCVFDT (Uncertainty-handling and Concept-adapting Very Fast Decision Tree) algorithm. Like CVFDT with the capability of detecting concept drift, UCVFDT works by keeping its model consistent with a sliding window of uncertain samples. Function G(:) denotes uncertain information gain. In [17] authors have proposed a Sensitive Concept Drift Probing Decision Tree algorithm (SCRIPT). The main contributions of SCRIPT are: a) it can avoid unnecessary system cost for stable data streams; b) it can efficiently rebuild classifier while data streams are instable; c) it is more suitable for the applications in which a sensitive detection of concept drift is required. our Sensitive Concept Drift Probing Decision Tree (SCRIPT) algorithm in this section. Based on the variation of CDAV (Class Distribution on the Attribute Value), SCRIPT aims to apply to large scale and high speed applications

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

which also require the sensitiveness to handle the drifting concepts. SCRIPT cuts the data stream into sequential data blocks. When the test threshold is set as α, each training data in the block should be read no more than once and processed in minimal constant time while concepts are stable. While concepts are instable, SCRIPT would detect concept drift, and then build the alternate tree to correct previously built tree. In [7] paper, Authors have proposed a novel method for on-line learning regression trees with option nodes (ORTO) from data streams. It is basically works on Hoeffding bound. Our method for learning Hoeffding-based option trees for regression addresses the problem of instability of tree learning, commonly seen in the case of highly correlated or equally discriminative attributes, i.e., in tie situations. Hoeffding trees can suffer from a delay in the learning process in such tie situations, because they assume that the data is in abundance and will never stop to stream in: Decisions on split selection are postponed resulting in lower learning rates. They show that option nodes are a natural and effective solution to the problem of dealing with multiple equally discriminative attributes (the tie problem). The additional structure of the option trees provides interesting and useful information on the ambiguity of the splits and thus on the existence of several equally relevant attributes. In [18] authors have demonstrated the efficacy of incorporating multiple paths via option nodes in Hoeffding trees (HOT). They described a method for controlling tree growth, and determined a reasonable number of options to explore. In all but one of our datasets the additional structure improved the performance of the classifier. Option trees represent a useful middle ground between single trees and ensembles. At a fraction of the memory cost an option tree can provide comparable accuracy performance and superior prediction speed which are important factors in data stream processing.

IV. COMPARATIVE ANALYSIS

Here we have shown some comparison of results between 4 algorithms. Here the comparative results of runtime in milliseconds shown in following table I from [3] [4] [7] [14] [15]. The table II shows the average classification error i.e. mean squared error. The following table shows the memory requirement of the algorithms in MB. Mean square error is less in FIOT as compared to others and memory requirement is different datasets where ORTO is having low memory requirement for LED and Waveform datasets while more for elevators and wine quality datasets. Table IV showing the comparative analysis of the different algorithms.

Table I: Runtime in Milliseconds

Dataset	OVA	UFFT	ORTO	FIMT-DD
LED	8060	37740	573.61	27110
Waveform	4850	7110	2625.1	26930

Table II: Mean squared error

Dataset	ORTO	FIMT-DD	FIOT
Elevators	18.0e-6 ± 1.0e-6	25.0e-6 ± 2.0e-6	0.729 ± 0.019
Wine quality	0.57 ± 0.06	0.59 ± 0.07	0.855 ± 0.019

Table III: Memory requirements in MB

Dataset	FIMT-DD	FIOT	ORTO	UFFT
LED	35.54	42.67	2.76	49
Waveform	35.54	48.32	12.27	116
Elevators	23.10	53.2	850.7	315
Wine quality	11.70	48.5	147.6	315

Table IV: Comparative Analysis of Incremental learning using Decision trees approaches

PARAMETERS	APPROACHES FOR INCREMENTAL LEARNING USING DECISION TREES				
	STREAMED DATA	TIME CONSUMPTION	DRIFT DETECTION	NEW ATTRIBUTE ARRIVAL	MEMORY REQUIREMENT
OVA (2009)	YES	HIGH	YES	YES	LOW
I+LEARNING (2008)	NO	LOW	NO	YES	LOW
UFFT (2005)	YES	LOW	YES	NO	HIGH
FIMT-DD (2010)	YES	LOW	YES	NO	HIGH
FIOT (2010)	YES	LOW	NO	NO	LOW
ORTO (2011)	YES	LOW	NO	NO	LOW
HOT (2007)	YES	LOW	NO	NO	LOW

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)

V. LIMITATION OF EXISTING APPROACHES

In Ova classifiers [13], as it is ensemble approach it is very time consuming. In [12], algorithm is able to handle the incremental learning regarding the new incoming attribute, does not work for streamed data. In [14] very large size decision tree, as it maintains naïve Bayes classifier in the intermediated nodes so due to both memory requirement is very high. Pruning of sub tree leads to catastrophic forgetting. In [4]Explicit drift detection, fast, stores statistics at leaves only so memory requirement is low for storing statistics, Generates large sized decision tree. In [3] first time option nodes are introduced in order to improve the bias management, introduces ambiguous splits under gradual drifts, lower classification accuracy and lacks drift detection. In [7] it works in tie situation, compact tree size, lacks drift detection and only applicable for regression that is the continuous values. In [18] it lacks drift detection; this is applicable for continuous as well as categorical.

VI. RESEARCH CHALLENGES

- In data stream mining arrival of novel class or novel label is a challenging problem
- The Multilabel classification is another challenging problem in the data stream mining.
- Memory requirement and the time required are quite high for streamed data, reducing them could a challenge because it generates a large tree which requires more memory, so to develop algorithm such a way that it generates the compact tree.
- Concept drift is problem in streamed data as the stream is continuous arriving data so concept changes time to time.
- In [12] the algorithm though is for incremental learning, I+Learning model can be extendable for classifying multi-label class problem, in which an instance belongs to multiple classes.
- In [4], challenge is to extend the algorithms for learning option trees (for regression) from time-changing streaming data and the problem of on-line computation of a confidence in individual predictions.
- In [7], perform a more thorough analysis of the overall performance of the system, including a comparison with ensemble methods, such as on-line bagging and random forests methods for regression The latter would be of special interest since the option trees with averaging can be viewed also as an ensemble method.
- In [3], we would like to investigate more strategies for combining predictions or choosing the best predictor. Another interesting problem is to employ the change detection mechanism over the different prediction strategies and in that way enable dynamic real-time decision-making mechanism that can suggest which predictions should be used at a given point in time. Another interesting problem is to employ the change detection mechanism over the different prediction strategies and in that way enable dynamic real-time decision-making mechanism that can suggest which predictions should be used at a given point in time. Another plan is to evaluate ideas on non-stationary real-world streaming problems, where an interpretable model can be useful.

VII. CONCLUSION

ORTO and HOT we find are better compared to others for incremental learning for streamed data. As both lack the drift detection mechanism so this could be the future work to be extendible.

ACKNOWLEDGMENT

I would like to take this opportunity to express my gratitude and heartiest thanks to Associate Prof. Amit Thakkar, Department of Information Technology, CSPIT, CHARUSAT, CHANGA, for his encouragement and guidance in the fulfillment of my project. He had abundantly helped and offered invaluable assistance, support and guidance. I wish to express my cordial appreciation to him for his valuable instructions and suggestions throughout this course.

I am also thankful to my head of department Prof. Amit Ganatra, head of the department, U & P.U patel department of computer engineering, CSPIT, CHARUSAT , CHANGA.I am also thankful to other faculty members of their help whenever I needed it.

REFERENCES

1. Mi losz R. Kmieciak and Jerzy Stefanowski Stream Handling Sudden Concept Drift in Enron Messages Data Mat. III KNTPD Conf., Poznan 21-23 April 2010, WNT Press, 2010, 284-296
2. Jose´ del Campo-A´ vila, Gonzalo Ramos-Jime´nez, Jo.ao Gama, and Rafael Morales-Bueno Improving Prediction Accuracy of an Incremental Algorithm Driven by Error Margins Intelligent Data Analysis - Knowledge Discovery from Data Streams Volume 12 Issue 3, August 2008
3. Elena Ikonomovsk, Jo˜ao Gama, Sa˘so D˘zeroski1 and Jozef Stefan Incremental Option Trees for Handling Gradual Concept Drift European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML PKDD 2010).
4. Elena Ikonomovska · João Gama · Sašo Džeroski Learning model trees from evolving data streams Discovery Science: Data Mining and Knowledge Discovery Volume 23 Issue 1, July 2011, 52-63
5. Mohammad M. Masud, Jing Gao, Latifur Khan, Classification and Novel Class Detection in Concept-Drifting Data Streams under Time Constraints Knowledge and Data Engineering, IEEE Transactions on Volume: 23 Issue:6 On page(s): June 2011, 859-874
6. Cheng-Jung TSAI Chien-I LEE Wei-Pang YANG An Efficient and Sensitive Decision Tree Approach to Mining Concept-Drifting Data Streams INFORMATICA, 2008, Vol. 19, No. 1, 2008, 135–156
7. Elena Ikonomovska Jo˜ao Gama Bernard ˇZenko Sa˘so D˘zeroski Speeding Up Hoeffding-Based Regression Trees with Options Appearing in Proceedings of the 28 th International Conference on Machine Learning ICML 2011: 537-544
8. Chunquan Liang Yang Zhang Qun Song Decision Tree for Dynamic and Uncertain Data Streams JMLR: Workshop and Conference Proceedings 13: 209-224 2nd Asian Conference on Machine Learning (ACML2010), 2010.

9. Induction of Decision Trees J.R. QUINLAN Mach. Learn. 1, 1 (Mar. 1986), 81-106
10. Jeffrey T. Byorick, Assesing Classification confidence using a weighted exponential based technique with the learn++ incremental learning algorithm, Lecture Notes in Computer Science Volume: 2714, 2002, pages 181-188
11. Albert Bifet. Adaptive learning and mining for data streams and frequent patterns. SIGKDD Explorations Newsletter, ACM, vol. 11, 2009, 55-56.
12. Sam Chao Fai Wong Yiping Li An Incremental and Interactive Decision Tree Learning Algorithm for a Practical Diagnostic Supporting Workbench Fourth International Conference on Networked Computing and Advanced Information Management IEEE, vol. 2, 2008 pages 202-207
13. Sattar Hashemi, Ying Yang, Zahra Mirzamomen, and Mohammadreza Kangavari Adapted One-versus-All Decision Trees for Data Stream Classification IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING,VOL. 21, MAY 2009 ,page No.. 5.
14. Jo˜ao Gama Pedro medas Learning Decision Trees from Dynamic Data Streams.2nd Asian Conference on Machine Learning (ACML2010),Journal of Universal Computer Science, ACM, vol. 11,2005, pages 1353-1366.
15. Dariusz Brzezinski MINING DATA STREAMS WITH CONCEPT DRIFT, master's thesis, Poznan, 2010
16. Chunquan Liang, Yang Zhang, Qun Song Decision Tree for Dynamic and Uncertain Data Streams JMLR: Workshop and Conference Proceedings 13: 209-224 2nd Asian Conference on Machine Learning (ACML2010), Tokyo, Japan, Nov. 8{10, 2010.
17. Cheng-Jung TSAI, Chien-I LEE, Wei-Pang YANG An Efficient and sensitive decision Tree Approach to mining concept drifting data streams INFORMATICA, 2008, Vol. 19, No. 1, 135–156
18. Bernhard Pfahringer, Geoffrey Holmes, and Richard Kirkby New Options for Hoeffding Trees Springer-Verlag Berlin Heidelberg M.A. Orgun and J. Thornton (Eds.): AI 2007, LNAI 4830,2007, pp. 90–99

AUTHOR PROFILE

Prerana Gupta has received her B.E degree in Computer Engineering from Veer Narmad South Gujarat University, Gujarat, India in 2009 and pursuing master degree from CHARUSAT since September 2010. Her Current Research work is in streamed datamining

Amit Thakkar has received his B.E degree in Information Technology from Gujarat University, Gujarat, India in 2002 and master Degree from Dharmsinh Desai University, Gujarat, India in 2007. He has joined his Ph.D in the area of Multi relational Classification at Kadi Sarva vishvidhalaya University, Gandhinagar, India in June 2010.

Since 2002 he has been with faculty of Engineering and Technology, Charotar University of Science and Technology, Changa, Gujarat, Where he is currently working as an Associate Professor in the Department of Information Technology. He has published more than 20 research papers in the field of data mining and web technology. His current research interest includes Multi relational Data Mining, Relational Classification and Associate Classification.

Amit Ganatra has received his B.E degree in Computer Engineering from Gujarat University, Gujarat, India in 2000 and master Degree from Dharmsinh Desai University, Gujarat, India in 2004. He has joined his Ph.D in the area of Multiple Classifier System (Information Fusion) at Kadi Sarvavishvidhalaya University, Gandhinagar, India in August 2008.

Since 2000 he has been with faculty of Engineering and Technology, Charotar University of Science and Technology, Changa, Gujarat, Where he is currently working as an Associate Professor in the Department of Computer Engineering. He has published more than 50 research papers in the field of data mining and Artificial Intelligence. His current research interest includes Multiple Classifier System, Sequence Pattern Mining

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)