Accuracy Extended Ensemble – A Brood Purposive Stream Data mining

Mr.Indrajitsinh S. Solanki
M.Tech Student of Department of Computer Science& Eng.
Kautilya Institute of Technology and Engineering and
School of Management(KITE-SOM),
Jaipur(Rajasthan),India.
inder_it88@yahoo.co.in

Prof.Tarannum.S.Bloch
Faculy of Engineering-IT,
Marwadi Education Foundation's Group of Instutions,
Rajkot(Gujrat),India.
taru.eng@gmail.com

Dr.Vijaykumar
Head of Department, Department of Computer Science & Engineering,
Kautilya Institute of Technology and Engineering and School of Management(KITE-SOM),
Jaipur(Rajasthan),India.
vijay_matwa@yahoo.com

Abstract— **The objective of Data mining is to haul out knowledge from gigantic quantity of data. The storage, querying and mining of such data sets are highly computationally challenging tasks. Mining data streams is concerned with extracting knowledge structures represented in models and patterns in non stopping streams of information. The research in data stream mining has gained a high attraction due to the importance of its applications and the increasing generation of streaming information. Decision trees have been widely used for online learning classification. In this article the problem of data-stream classification has been considered by introducing an online and incremental stream-classification ensemble algorithm given name Accuracy Extended Ensemble which an extension to the Accuracy Weighted Ensemble. Proposed algorithm will be adept to deal with data streams having an evolving nature and an ergodic arrival rate of training/test data records.**

Index Terms — Accuracy, Data Stream Mining, Ensemble Technique, Hoeffding bound, Tree Bagging.

*****

I. INTRODUCTION.

In today's information society, computer users are used to gathering and sharing data anytime and anywhere. This concerns applications such as social networks, banking, telecommunication, healthcare, research, and entertainment, among others. As a result, a huge amount of data related to all human activity is gathered for storage and processing purposes. These data sets may contain interesting and useful knowledge represented by hidden patterns, but due to the volume of the gathered data it is impossible to manually extract that knowledge. That is why *data mining* and *knowledge discovery* methods have been proposed to automatically acquire interesting, non-trivial, previously unknown and ultimately understandable patterns from very large data sets [1, 2].

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 [3]:

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.

Data streams can be viewed as a sequence of relational tuples (e.g.,call records, web page visits, sensor readings) that arrive continuously at time-varying, possibly unbound streams. Due to their speed and size it is impossible to store them permanently [4]. Data stream application domains include network monitoring, security, telecommunication data management, web applications, and sensor networks. The introduction of this new class of applications has opened an interesting line of research problems including novel approaches to knowledge discovery called *data stream mining*.

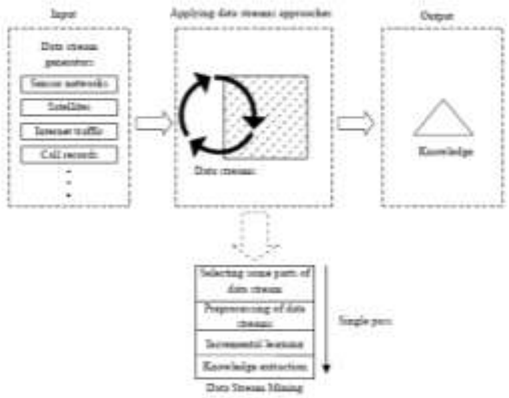

Fig 1: General process of data stream mining.

II. DATA STREAM MINING APPROACH

Recent research addresses the problem of data-stream mining to deal with applications that require processing huge amounts of data such as sensor data analysis and financial applications. Data-stream mining algorithms incorporate special provisions to meet the requirements of stream-management systems, that is stream algorithms must be online and incremental, processing each data record only once (or few times); adaptive to distribution changes and fast enough to accommodate high arrival rates.

Because of above described characteristics of data stream mining it cannot be handled by traditional data mining approach. Table-1 lists out difference between traditional data mining & data stream mining.

TABLE I. COMPARISONS TRADITIONAL & STREAM DATA MINING[5]

	Traditional Data Mining	**Data Stream Mining**
No. of passes	Multiple	Single
Processing time	Unlimited	Restricted
Memory usage	Unlimited	Restricted
Type of result	Accurate	Approximate
Concept	Static	Evolving
Distributed	No	Yes

Data stream mining can be performed by single classier or ensemble approach. Some of the popular single classifiers proposed for stationary data mining fulfill both of the stream mining requirements have the qualities of an online learner and a forgetting mechanism. Some methods that are only able to process data sequentially, but do not adapt, can be easily modified to react to change. These classifiers fall into these groups: neural networks, Naive Bayes, nearest neighbour methods, and decision rules.

Classifier ensembles are another common way of boosting classification accuracy. Due to their modularity, they also provide a natural way of adapting to change by modifying ensemble members. Ensemble algorithms are sets of single classifiers (components) whose decisions are aggregated by a voting rule. The combined decision of many single classifiers is usually more accurate than that given by a single component. Studies show that to obtain this accuracy boost, it is necessary to diversify ensemble members from each other. Components can differ from each other by the data they have been trained on, the attributes they use, or the base learner they have been created from. For a new example, class predictions are usually established by member voting. Kuncheva [6] proposes to group ensemble strategies for changing environments as follows:

- *Dynamic combiners (horse racing)* - individual classifiers (experts) are trained in advance and the forgetting process is modeled by changing the expert combination rule.
- *Updated training data* - the experts in the ensemble are created incrementally by incoming examples. The combination rule may or may not change in the process.
- *Updating the ensemble member* - ensemble members are update online or retrained with blocks of data.
- *Structural changes of the ensemble* - periodically or when change is detected, ensemble members are reevaluated and the worst classifiers are updated or replaced with a classifier trained on the most recent examples.
- *Adding new features* - as the importance of features evolves with time, the attributes used by team members are changed without redesigning the ensemble structure.

There are many ensemble classifier algorithm exists namely : Streaming Ensemble algorithm, Hoeffding option trees (HOT) and Adaptive-Size Hoeffding Tree Bagging (ASHT) Bagging, Accuracy Weighted Ensemble (AWE). We are going to elaborate AWE here.

ACCURACY WEIGHTED ENSEMBLE

A new way of restructuring an ensemble was proposed by Wang et al. [7]. In their algorithm, they train a new classifier C’ on each incoming data chunk and use that chunk to evaluate all the existing ensemble members to select the best component classifiers.

```
Algorithm: Accuracy Weighted Ensemble
Input:      S: a data stream of examples
            d: size of data chunk xi
            k: the total number of classifiers
            C: a set of previously trained classifiers
Output: Ɛ : a set of k classifiers with updated weights
            for all data chunks xi ϵ S do
            train classifier C’ on xi;
            compute error rate of C’ via cross validation on S;
```

```
derive weight w' for C' using equation $w_i = MSE_r - MSE_i$;
for all classifiers Ci ϵ C do
apply Ci on xi to derive MSEi;
compute wi based on euation $w_i = MSE_r - MSE_i$;
Ɛ ← k of the top weighted classifiers in C ∪ {C'}
    C ← C ∪ {C'}
```

AWE combines multiple classifiers weighted by their expected prediction accuracy on the current test data. Compared with incremental models trained by data in the most recent window, their approach combines talents of set of experts based on their credibility and adjusts much nicely to the underlying concept drifts. Also, they introduced the dynamic classification technique to the concept-drifting streaming environment, and results show that it enables to dynamically select a subset of classifiers in the ensemble for prediction without loss in accuracy.

Authors use weighted ensemble classifiers on concept-drifting data streams (the distribution generating the items of a data stream can change over time. These changes, depending on the research area, are referred to as *temporal evolution*, *covariate shift*, *non stationarity*, or *concept drift* [7]).

III. Problem formulation

Drawback of AWE is its weighting function. Because the algorithm is designed to perform well on cost-sensitive data, the Mean Square Error threshold in cuts-off risky classifiers. Another problem is processing of data blocks allowed AWE to learn component classifiers by traditional batch algorithms (not special online ones) and later only adjust component weights according to the current distribution. To have solution of above described problem we have bespoke AWE algorithm.

IV. Proposed Algorithm

In rapidly changing environments with concept drifts threshold can "mute" all ensemble members causing no class to be predicted. To avoid this, in we propose a simpler weighting function. This weighting function was originally used in Hoeffing Tree Algorithm [8] which is single classifier approach. So we are changing weighting function of Accuracy Weighted Ensemble by applying following weighing function :

$$w_i = \frac{1}{(MSE_i + \epsilon)}$$

Here ϵ is Hoeffding bound, very small integer constant value calculated using.

$$\epsilon = \sqrt{\frac{R^2 \ln(1/\delta)}{2n}}.$$

Input: S: data stream of examples
k: number of ensemble members
Output: E: ensemble of k online classifiers with newly assigned weights

Process :

```
        C ← θ: set of stored classifiers

        for all data chunks $x_i$ S do
                train classifier $C_0$ on $x_i$;
                compute error MSE of $C_0$ via cross
        validation on $x_i$;
                derive weight w' for C' using $w_i$;
for all classifiers $C_i$ ϵ C do
        apply $C_i$ on $x_i$ to derive $MSE_i$;
        compute weight $w_i$ ;
        E ← k of the top weighted classifiers in C ∪
                {C'};
        C ← C ∪ {C'};
for all classifiers $C_e$ ϵ ϵ do
        if $w_e > 1/MSE_r$ and $C_e \neq C'$ then update
classifier $C_e$ with $x_i$;
```

STREANTH OF PROPOSED ALGORITHM

We propose a new adaptive ensemble and named it as Accuracy Extended Ensemble (AE2).The proposed algorithm is inspired by AWE and its weighting mechanism. In AE2 we turn to online learning of component classifiers. This allows updating base classifiers rather than only adjusting their weights. The Hoeffding bound [8] states that with probability 1 - δ, the true mean of a random variable of range *R* will not differ from the estimated mean after *n* independent observations by more than

$$\epsilon = \sqrt{\frac{R^2 \ln(1/\delta)}{2n}}.$$

This bound is useful because it holds true regardless of the distribution generating the values, and depends only on the range of values, number of observations and desired confidence.

V. Experimental enviornment

Massive Online Analysis (MOA) is a software environment for implementing algorithms and running experiments for online learning [9, 10, 11]. It is implemented in Java and contains a collection of data stream generators, online learning algorithms, and evaluation procedures. Currently we are implementing proposed algorithm AE2 (Accuracy Extended Algorithm) and will be compare with other implemented algorithm of MOA framework. All the other algorithms are already a part of MOA. The experiments took place on a machine equipped with an Intel core i3 M 370 @ 2.40 GHz processor and 3.00 GB of RAM. Each algorithm will be tested on different data sets using the Data Chunk evaluation procedure

VI. Conclusion.

Presently we are implementing Accuracy Extended Ensemble (proposed algorithm) using MOA framework & it will be tested for unconsidered parameters. The weaknesses of Accuracy Weighted Ensemble classifier will be detached by applying Hoeffding bound in weighting function. This will be helpful to increase the accuracy of the AE2 classifier. Hoeffding bound has been applied to many single classifiers and many ensemble algorithms. Hoeffding bound has given pretty good results in all these classifiers. So we are quite sure about accuracy pay raise with our algorithm. In the case of collapse we can apply other bound such churnoff's bound, markov's inequality, chebyshev's bound, benette's bound etc can be applied to the weighting equation. Partial realization with hoeffding bound is completed, will get statistical consequences soon.

Acknowledgment

This research is the part of M.Tech. Programme, at Rajashthan Technological University, Rajashthan, India.The MOA (Massive Online Analysis) a project of The University of Waikato is the source of the framework and its assistance is gratefully acknowledged.

References

[1] Usama M. Fayyad, Gregory Piatetsky-Shapiro, and Padhraic Smyth. From data mining to knowledge discovery: An overview. In *Advances in Knowledge Discovery and Data Mining*, pages 1–34. American Association for Artificial Intelligence, 1996.

[2] Max Bramer. *Principles of Data Mining*. Springer, 2007.

[3] Albert Bifet. *Adaptive learning and mining for data streams and frequent patterns*. PhD thesis,Universitat Polit´ecnica de Catalunya, 2009.

[4] Elena Ikonomovska, Suzana Loskovska, and Dejan Gjorgjevik. A survey of stream data mining,2005.

[5] Mohamed Medhat Gaber and Jo˜ao Gama. State of the art in data streams mining. ECML,2007.

[6] Ludmila I. Kuncheva. Classifier ensembles for changing environments. In Fabio Roli, Josef Kittler, and Terry Windeatt, editors, *Multiple Classifier Systems*, volume 3077 of *Lecture Notes in Computer Science*, pages 1–15. Springer, 2004.

[7] Haixun Wang, Wei Fan, Philip S. Yu, and Jiawei Han. Mining concept-drifting data streams using ensemble classifiers. In Lise Getoor, Ted E. Senator, Pedro Domingos, and Christos Faloutsos, editors, *KDD*, pages 226–235. ACM, 2003.

[8] G. Hulten, L. Spencer, and P. Domingos. Mining time-changing data streams. In *KDD'01*, pages 97–106, San Francisco, CA, 2001. ACM Press.

[9] Albert Bifet, Geoff Holmes, Richard Kirkby, and Bernhard Pfahringer. Moa: Massive online analysis. *Journal of Machine Learning Research*, 11:1601–1604, 2010.

[10] Albert Bifet and Richard Kirkby. *Massive Online Analysis*, August 2009.

[11] Albert Bifet and Richard Kirkby. Data stream mining: a practical approach. Technical report, The University of Waikato, August 2009.