International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 05 Issue: 08 | Aug 2018 www.irjet.net p-ISSN: 2395-0072

A data stream mining technique dynamically updating a model with dynamic changes of data distributions

Mehathaj Kathu S

[1]*Mehathaj Kathu S Address: Thanjavur*
[2]*Professor: Mrs. R. INDRA,M.Sc.,M.Phil.,MCA.,Dept. of Computer Science, Shrimathi Indira Gandhi College, Tamil Nadu, India*

Abstract - Data streams, which can be considered as one of the primary sources of what is called big data, arrive continuously with high speed. The biggest challenge in data streams mining is to deal with concept drifts, during which ensemble methods are widely employed. The ensembles for handling concept drift can be categorized into two different approaches: online and block-based approaches. The primary disadvantage of the block-based ensembles lies in the difficulty of tuning the block size to provide a tradeoff between fast reactions to drifts. Motivated by this challenge, we put forward an online ensemble paradigm, which aims to combine the best elements of block-based weighting and online processing. The algorithm uses the adaptive windowing as a change detector. Once a change is detected, a new classifier is built replacing the worst one in the ensemble. By experimental evaluations on both synthetic and real-world datasets, this method performs significantly better than other ensemble approaches.

Key Words: **Data Mining, Change Detection, Concept Drift**

1. INTRODUCTION

In recent years, some promising computing paradigms have emerged to meet the needs of big data. The only thing that the parallel batch process model copes with is the stationary massive data. However, there are a lot of applications in practice, such as sensor networks, spam filtering, intrusion detection, and credit card fraud detection, which generate continuously arriving data, known as data streams. Most big data can be regarded as data streams, in which data are produced continuously. In fact, model in the data stream is coping with the problem of three features of big data: big volume, big velocity, and big variety. According to their speed, concepts drifts have been divided into two types: sudden drifts and gradual drifts. Sudden concept drift is characterized by large amounts of change between the underlying class distribution and the incoming instances in a relatively short amount of time, while gradual concept drift is featured by large amount of time to witness a significant change in differences between the underlying class distribution and the incoming instances. Most of the existing methods just deal with one of the two types. However, in the real-world, data stream probably contains more than one type of concept drift. Thus, being able to track and adapt to various kinds of concept drift instantly is highly expected from a better classifier. The performance of the proposed algorithms was evaluated on both synthetic and real-world datasets, and a comprehensive comparison study of online and block-based ensemble algorithms was presented. The results show that this method achieves better performance than previous methods, especially when concept drift occurs.

2. Existing System

Most of the existing solutions constructing stream data mining are under the hypothesis that data are stationary. However, in the real-world, the generation of data streams is usually in the non stationary environment, which means that the underlying distribution of the data can change arbitrarily over time. This phenomenon is known as concept drift, which exists commonly in the scenarios of big data mining. For example, weather prediction models change according to the seasons, and in recommend systems, user consumption patterns may change over time due to fashion, economy, and so forth. The occurrence of such change leads to a drastic drop in classification accuracy. Therefore, the learning models should be able to adapt to the changes quickly and accordingly.

2.1 Survey

A. Incremental Ensemble Classifier Addressing Non-stationary Fast Data Streams

Classification of data points in a data stream is a fundamentally different set of challenges than data mining on static data. While streaming data is often placed into the context of "Big Data" (or more specifically "Fast Data") wherein one-pass algorithms are used, true data streams offer additional hurdles due to their dynamic, evolving, and non-stationary nature. During the stream, the available labels (or concepts) often change, and a concept's definition in the feature space can also evolve (or drift) over time. The core issue is that the hidden generative function of the data is not a constant function, but rather evolves over time. This is known as a non-stationary distribution. In this paper, describe a new approach to using ensembles for stream classification. While the core method is straightforward, it is specifically designed to adapt quickly with very little overhead to the dynamic and evolving nature of data streams generated from non-stationary functions. This method, M^3, is based on a weighted majority ensemble of

heterogeneous model types where model weights are updated on-line using Reinforcement Learning techniques. Compare this method with current Leading algorithms as implemented in the Massive Online Analysis (MOA) framework using UCI benchmark and synthetic stream generator data sets, and find that this method shows particularly strong gain over the baseline method when ground truth is of limited availability to the classifiers.

B. CD-TDS: Change detection in transactional data streams for frequent pattern mining

Online mining is a difficult task especially when such data streams evolve over time. Evolving data stream occurs when concepts drift or change completely, is becoming one of the core issues. A large portion of change detection research are carried out in the area of supervised learning, very little has been carried out for unlabeled data specifically in the area of transactional data streams. Overall when the monitor changes in transactional data can consider two different types of changes: local and global change. Local changes are changes in distribution of the data, whereas global changes are data composition changes within the data stream. To detect changes in transactional data streams containing unlabeled data, introduce a new technique called CD-TDS that detects both these changes. The change detector can identifies changes in relationships between items as data evolves with the progression of a stream. Crucially, detection of global drift enables us to better understand the dynamics in relationships that takes place over time. Experimental results using both real world and synthetic data show that the proposed approach is robust to noise and identifies structural changes with a high true positive rate while preserving a low false alarm rate.

C. Efficient handling of concept drift and concept evolution over Stream Data

To decide if an update to a data stream classifier is necessary, existing sliding window based techniques monitor classifier performance on recent instances. If there is a significant change in classifier performance, these approaches determine a chunk boundary, and update the classifier. However, monitoring classifier performance is costly due to scarcity of labeled data. In previous work, presented a semi-supervised framework SAND, which uses change detection on classifier confidence to detect a concept drift. Unlike most approaches, it requires only a limited amount of labeled data to detect chunk boundaries and to update the classifier. However, SAND is expensive in terms of execution time due to exhaustive invocation of the change detection module. In this paper, present an efficient framework, which is based on the same principle as SAND, but exploits dynamic programming and executes the change detection module selectively. Moreover, we provide theoretical justification of the confidence calculation, and show effect of a concept drift on subsequent confidence scores. Experiment results show efficiency of the proposed framework in terms of both accuracy and execution time.

D. Big-data streaming applications scheduling with online learning and concept drift detection

Several techniques have been proposed to adapt Big-Data streaming applications to resource constraints. These techniques are mostly implemented at the application layer and make simplistic assumptions about the system resources and they are often agnostic to the system capabilities. Moreover, they often assume that the data streams characteristics and their processing needs are stationary, which is not true in practice. In fact, data streams are highly dynamic and may also experience concept drift, thereby requiring continuous online adaptation of the throughput and quality to each processing task. Hence, existing solutions for Big-Data streaming applications are often too conservative or too aggressive. To address these limitations, propose an online energy-efficient scheduler which maximizes the QoS (i.e., throughput and output quality) of Big-Data streaming applications under energy and resources constraints. scheduler uses online adaptive reinforcement learning techniques and requires no offline information. Moreover, scheduler is able to detect concept drifts and to smoothly adapt the scheduling strategy. Experiments realized on a chain of tasks modeling real-life streaming application demonstrate that scheduler is able to learn the scheduling policy and to adapt it such that it maximizes the targeted QoS given energy constraint as the Big-Data characteristics are dynamically changing.

E. Using count prediction techniques for mining frequent patterns in transactional data streams

In this system study the problem of mining frequent itemsets in dynamic data streams and consider the issue of concept drift. A count-prediction based algorithm is proposed, which estimates the counts of itemsets by predictive models to find frequent itemsets out. The predictive models are constructed based on the data in the data stream and serve as a description of the concept of the stream. If there is a concept drift in the stream, the description of the concept can be updated by reconstructing the predictive models. According to our experimental results, the proposed algorithm is efficient and has stable performance. Besides, using respective predictive models for count-predictive mining would preserve the quality of mining answers effectively (in terms of accuracy) against the change of the concept.

3. Proposed System

Concept drift has become a popular research topic over the last decade and many algorithms have been developed. The methodologies proposed for tackling concept drifts can be organized into three main groups: window based approaches, weight-based approaches, and ensemble classifiers. Ensemble methods are widely used in concept

drift learning. The techniques for using ensemble to handle concept drift fall into two categories: block-based ensembles and online ensembles. For block-based ensembles, the streams are segmented into a series of successive fixed-size blocks. Online ensembles update component weights after each instance without the need for storage and reprocessing. So this method can adapt to sudden changes as quickly as possible. However, some of these algorithms are usually characterized by higher computational costs compared with block-based methods.

Change Detector Algorithm

Change detection and notification (CDN) refers to automatic detection of changes made to World Wide Web pages and notification to interested users by email or other means. Whereas search engines are designed to find web pages, CDN systems are designed to monitor changes to web pages. Before change detection and notification, it was necessary for users to manually check for web page changes, either by revisiting web sites or periodically searching again. Efficient and effective change detection and notification is hampered by the fact that most servers do not accurately track content changes through Last-Modified or ETag headers.

Input: data stream S, confidence $\delta \in (0, 1)$;
Output: ChangeAlarm;
(01) Initialize Window W;
(02) **for** each $t > 0$ do
(03) $W \leftarrow W \cup \{x_t\}$ (i.e., add x_t to the head of W);
(04) repeat
(05) Drop elements from the tail of W_L;
(06) **until** $\mathrm{KL}(W_L \parallel W_R) < \varepsilon$; (calculate ε according to (4));
(07) end for
(08) Output ChangeAlarme;
(09) end

Algorithm 1: Pseudocode of adaptive windowing change detector.

Classification Accuracy

In terms of accuracy Level and our method outperform all the other algorithms. On the dataset with no drift (Waveform), Lev, AWE, and DWM performed almost identically, with OAUE being slightly less accurate. For the dataset with gradual concept drift (HyperPlane), AWE is the best, followed by AUE. However, our method seems to be the most accurate in the case of sudden changes (SEA). This is partly because the addition of drift detector offers quicker reactions to sudden concept changes compared to most block-based ensembles. For the dataset with mixed concept drift (LED), our proposed method largely outperformed other algorithms. On the real world datasets, in terms of accuracy, there is no single best performing algorithm. On the Covertype, our method clearly outperformed all the other algorithms. On the Poker, OAUE is the most accurate followed by Level, while on the Electricity all the algorithms perform almost identically.

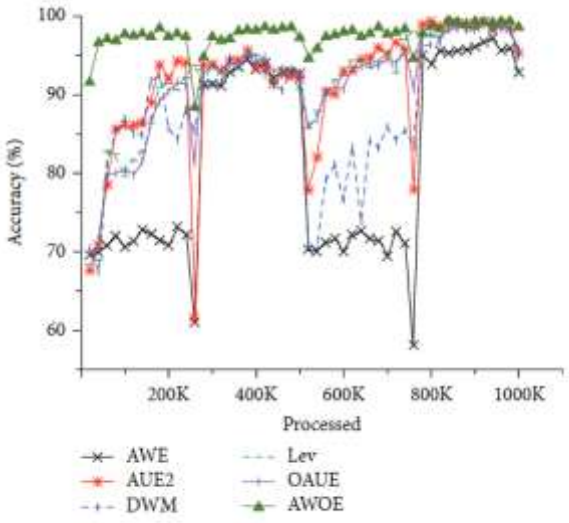

FIGURE 1: Accuracy on the SEA dataset.

4. Conclusion

This study, through studying the influence of the size of data block on performance of the ensemble classifier, proposed an online ensemble with internal change detector to capture concept drifts in timely manner by determining block size dynamically. The experimental results prove that our approach performs better than other ensembles and gains the best tradeoff between accuracy and resources. Most existing data stream algorithms assume that true labels are immediately and entirely available. Unfortunately, such assumption is often violated in real-world applications because it is expensive to obtain all true labels.

REFERENCES

[1] Real-time data mining of non-stationary data streams from sensor networks- L. Cohen, G. Avrahami-Bakish, M. Last, A. Kandel, and O. Kipersztok.

[2] A case based technique for tracking concept drift in spam filtering- S. J. Delany, P. Cunningham, A. Tsymbal, and L. Coyle.

[3] Approaches to online learning and concept drift for user identification in computer security- T. Lane and C. E. Brodley.

[4] Mining concept drifting data streams using ensemble classifiers- H. Wang, W. Fan, P. S. Yu, and J. Han.

[5] Data Streams: Models and Algorithms- Springer, Berlin, Germany, C. C. Aggarwal.