ISSN: 2277-9655
(I2OR), Publication Impact Factor: 3.785

IJESRT

INTERNATIONAL JOURNAL OF ENGINEERING SCIENCES & RESEARCH TECHNOLOGY

AN EFFICIENT INDEXED BASED CLASSIFICATION APPROACH FOR FEATURE EVOLVING DATA STREAMS

Jyoti Wagde*, Prarthana Deshkar

* Department of Computer Technology ,Yeshwantrao Chavan College of Engineering, Nagpur, India

DOI: 10.5281/zenodo.49791

ABSTRACT

Today, rapid growth in hardware technology has provided a means to generate huge volume of data continuously. In most of the real time data stream application data usually reach very rapidly that flows continuously in real time environment .This incoming data streams comprises of several important and interesting patterns underneath. However, mining an essential data out from this data stream, has some major challenges such as infinite length, concept evolution and concept-drift .Earlier studies, carried till now, have been mainly focusing on building accurate classification model. But, keeping glance over the huge amount of classifiers while prediction, require more response time, which makes ensemble approach being impractical for many real-world times critical data stream applications. In order to over this problem, we propose a new enhanced indexing structure that organizes all classifiers of ensemble in order to get fast prediction response in lesser amount of time .In addition to this, ensemble model is updated continuously by integrating new classifiers, while adapting to new trends. Experimental results and theoretical analysis on both real-worlds as well as synthetic data streams exhibit the better performance of our method over the existing techniques.

KEYWORDS: Data stream mining, concept drift, concept evolution, classification, ensemble.

INTRODUCTION

In recent years data stream classification has been a widely studied research problem in field of data mining and analysis. It has been popularly used in most of the real-time application such as, network traffic monitoring, spam filtering, malicious website monitoring, stock market fluctuations and credit card transactions [1] etc. These real time applications, contains millions of transactions, billions of operations and several records, which in turns creates large amount of data that flows continuously in real time environment and it comprises of various important and interesting patterns underneath hence classification is needed in order to get useful information out of this. The method of extracting information and knowledge from the endless data instances is called data stream classification [10]. The objective of data mining classifiers is to predict class label for new upcoming instance, whose attribute values are well-known in advanced but the class labels is not known that need to infer. It is also referred as supervised learning since the classes are determined before examining the new upcoming data and data is maps into training set. Due to certain distinctive properties shown by data streams classification of stream data become more difficult compared to traditional classification. Typically there are three major problems that appeared in this field. Primarily, data streams have huge, infinite length of data, which makes it impractical to store on the disk and even though if the data get stored on the disk it is unfeasible to scan the data recursively in order to identify the beneath patterns. Therefore, those algorithms which require multi-pass learning could not be appropriate to data streams, so we require an algorithm where stream data should use all the data while scanning only once. Secondly, concept drift is sense by data streams, which arises when the existing concept (values) of the data varies over time. In order to handle the concept-drift problem, classification model need quick response, to include alterations in data ,so that more accurate classification results could get reflect in further incoming stream. Thirdly, concept-evolution is also notice under the data streams, which basically appears when a novel class emerges in the stream. So, to overcome concept-evolution problem, a classification model should be enough capable to automatically identify novel classes

if they get found in the stream data, and need to integrate in the training model before the new data being classified. Furthermore feature evaluation is also there in the data stream, suppose if we have a static feature set, then we can merely use that available feature set but if the feature are dynamic and evolving, we need to an employ a feature selection as well as feature extraction technique [2]. The rest of the paper is structured as follows. Section II introduces the related work that has carried out in the field of data stream mining. Section III outlines the architecture and indexing structure for classification. Section IV reports the experimental analysis and result followed by the conclusion in Section V.

RELATED WORK

Classification of data stream instances has been an appealing research topic for several years, numerous methods and several approaches are available. All this approach comes under two categories: single model incremental approaches and ensemble classification approach [4]. Basically single model incremental approaches has fixed structure of the model which uphold and incrementally renew a single model, it has free-form functions to match the data .Along with this it also has some preference criteria such as information gain, gini-index. Whereas ensemble approach combines multiple models into one, which gives the global picture, here classification output is based on different classifiers. An ensemble classifier is created from several classifiers that independently acquire the class limits in a training set. The end result of an ensemble classifier on a data stream is generated by combining the distinct results of the base classifiers. Moreover, ensemble classifiers also recognized as compound classifier systems, group of classifiers as well as combination of experts. The foremost objective of ensemble learning is to form different classifiers. It has been seen that ensemble models are more popular as compared to single incremental model because of they are simpler to get implement, it addition to this, they also had shown greater efficiency, while adjusting quickly to new concepts, it also has ability to scale well, and provide the ease of parallelization [5]. So recently, various ensemble-based models have been proposed, in order to address the data stream challenges, such as classifier and cluster ensembles [7], incremental classifier ensembles [6] and weighted classifier ensemble, [8], [9], [10], [11]. Although these models vary from one to another, Most of these ensemble techniques share remarkable resemblance in their design approach: use of divide-and-conquer practice in which stream of data is divide into numerous chunks[3], and each data chunk is used in order to train a classification model ,which is capable to handle large amount of stream data along with concept evolution and concept drifting.

In past few years data stream analysis techniques are widely studied. Various types of analysis had made in the literature of data stream mining for different applications. Peng Zhang et. al. [3] in 2015,provived E-trees where they have treated ensembles as spatial databases then formerly applied spatial indexing techniques. R-tree like height-balanced structure is used in order to decrease the estimated prediction to sub-linear complexity time from linear time complexity. Moreover, E-trees can be mechanically restructured by constantly incorporating new classifiers and removing outdated ones, while acclimating to new trends and patterns beneath upcoming data streams.

Sattar Hashemi et. al. [13] in 2009, has proposed Adapted One-versus-All (OVA) Decision Trees for Data Stream Classification where k individual binary classifier is used and in order to classify a new instance, the k classifiers are run simultaneously and the one that yields the highest confidence is chosen. Since, there is low error correlation and high diversity among OVA's classifiers; it leads to high classification accuracy and greater performance.

Mohammad M. Masud et. al. [15] in 2011 describes a stream data classification, in which each classifier is constructed with a novel class detector, to overcome concept-drift and concept evolution problems. Feature set homogenization technique for feature-evolution. Enhanced novel class detection module identify more than one novel class at a same time which make it more robust and had improves the performance in terms of space, time and accuracy.

G.Divya et. al. [16] in 2014, has provided solution for the feature-evolving data streams they have proposed a hybrid method to detect and classify the novel class. For the detection novel class this methodology uses the Nearest Neighbor algorithm in addition with the Naive Bayes classifier concepts. Whenever a new feature appears in data stream, the old get removed and the new features get equipped. To eliminate the undesirable data on the data streams

outlier detection techniques are used so that the accurate results could get further.

PROPOSED WORK

Overview

Initially, we mathematically formulate the classification problem for data stream.

- Consider a two-class data stream **S,** which comprising infinite number of records {xi, yi}, where each xi is a n-dimension attribute vector ,x1 is the initial data attribute value in the stream, and xi is the last value that has arrived.
- Each data attribute value xi is associated with another attribute yi where yi is its class label, which will not get observed unless the incoming record is properly labeled.
- Suppose that we have constructed **m** base classifiers C1; C2; . . . ; Cn from historical stream data. All the **m** base classifies are collectively forms an ensemble classifier **E.**
- Now , using ensemble model **E** we are to predict the class label for an incoming stream record xi.

Ensemble Learning with Indexing

Fig.1. shows the architecture of ensemble learning with indexing on data streams. Here, training module keeps a classification model updated by repetitively calling the insertion where the classifiers are made based on available labeled data. Initially for each incoming stream record concept drift as well as concept evolution will be required to detect that can would be further used to build a new classifier and in order to keep the training model up-to-date. Here, the records are stored in a buffer for a while; afterwards it subsequently get labeled by experts (human experts) and later will be incorporated in training model. On the other hand, the prediction module will predict class label by using classification and learning operation

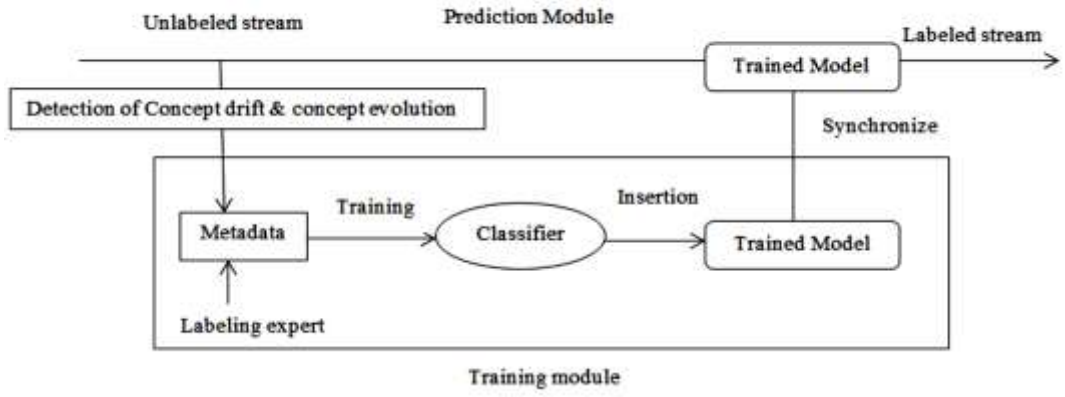

Fig.-1: Architecture of ensemble learning with indexing

Indexing structure for classification

The proposed method is used to classify the stream data, as well as to detect the concept drift and concept evolution in dynamic and feature evolving environment. The following section describes the comprehensive structure of the whole system. Our proposed method is divided into three parts: offline pre-training, online classification & learning.

1.Pre-Training

Offline pre-training is performed once at the start up to prepare the base classifier model. In this stage, trained data is extracted and then the metadata is created in database, the table structure in the database side will store the all information associated with classifier of the ensemble, such as IDs, attribute values and the class label for the attributes of classifiers. All the base classifiers information are denoted as follows

(classifier id; attrname; attrvalue;) (1)

where classifier ID is represented by the first item, second item represents the classifier's value, and the class label associated with the classifier values is denotes in the last item .For all the values which is present in the database class label is set manually which is used trained the data model and help in future in order to get the class label for new upcoming data stream.

2.Detection of concept drift and concept evolution

Let E is the current ensemble of classification model. A class C is an existing class if at least one of the models Ei ∈ E has been trained with the instances of class C. Otherwise, C is a detection of concept evolution. Along with this each occurrence of the most recent unlabeled chunk is examined by the ensemble models to check if it is outside the decision boundary of the ensemble. If it is within the decision boundary then, stream is classified based on the existing model otherwise this detected values is outline as concept evolution. When a new classifier C and new values is arrives in data stream, a new entry associated with C is added into the table structure in order to keep classification model up-to-date, so that the ensemble model can adapt to new trends and patterns in data streams.

3.Online Classification and Learning

Whenever a new record x arrives, each time a classify operation is called, in order to predict a class label for incoming record. The algorithm initially get pass through the database and finds decision rules inside the metadata that covers record x. From all the retrieved rules class label for x is then calculates with the given equation,

$$yx=\mathrm{sgn}\left(\max_{i=1}^{u}\left(\Sigma P(\mathrm{Cid},\mathrm{Cattrvalue}=\text{'yes'}),P(\mathrm{Cid},\mathrm{Cattrvalue}=\text{'no'})\right)\right) \quad (2)$$

where the class label of x is represented by yx, u denotes the total number of retrieved decision rules that covers x, sgn(a,β)is a threshold function which is decided by relating a and β that will further helps to select x's class label, and Cattrvalue is the class label in the metadata. To derive a class label for x, it essentially traverses along the database whose attrname cover x, and then calculates the class label for x using Eq. (2).

It has been noticed that there is a relationship between query cost and data dimensionality. The query cost increase depends on the number of attribute. In the worst case, the data are distributed in high-dimension, and then the query cost rises exponentially with the increase of data dimensionality. In the best case, if the data is densely distributed in a low-dimension, then the query cost increases linearly with respect to the increase of the dimensionality. Based on the above analysis, we study significant issues on the number of comparisons required for classifying x. In order to classify the newly arrived data stream it needs to break down into chunk and each divided chunk is glance over in database for their respective class label which will functionally require large database operation and hence increase the prediction time. So, the caching can be applied which can decrease the prediction time for classifying the similar record recursively.

EXPERIMENTAL ANALYSIS & RESULTS

Now under this section, we present extensive experiments on both synthetic and real-world data streams to validate the performance of indexing approach of classification with respect to prediction time, memory usage and prediction time with caching as well as without caching. All experiments are conducted on a Windows machine with 3 GHz CPU and 2 GB memory.

Benchmark data streams

A Real-world stream from the UCI repository [18] and one synthetic steam were used. For the malicious url detection dataset, only 14 attributes that are in discrete form are selected from the total of 31 attributes where it contains 48842 instances. The synthetic stream was generated as follows. First, we gather the mobile specification data from the GSM Arena and then randomly generated a collection of stream data {…..,(xi,yi),….} where xi ∈ Rd is a d-dimensional vector and yi ∈{yes, no} is the class label. Each stream record is defined by xi ≤ a , where a ∈ [0,n] here n is initially 18 attributes. To simulate concept drifting, after generating a data chunk of D records, we

randomly selected its jth dimension and randomly changed the values inside the attributes as well as added the new M dimension for generating the concept evolution.

Benchmark methods

We implemented three methods for the purpose of comparisons. 1) Ensemble: It has no upper bound on the number of base classifiers in classification model. For each new base classifier, it simply integrates it into the model and deletions are never invoked. 2) E-tree: Different from ensemble, E-tree has fix upper bound on the ensemble size. As soon as the upper bound is reached, the oldest classifier will be removed from the model. 3) Indexing approach: A linear scan is used during prediction. Classifiers will be added into the ensemble model continuously without deletions.

Measures

To evaluate the improve predicting efficiency of our approach, we compared the E-tree, Ensemble and Indx-App methods by varying ensemble size n. From the results in Fig. 2 and 3, we can come to two important conclusions: 1) Time cost: By using a indexing for all classifiers in classification we are expected to attain a much lower computational cost than previously available ensemble models. 2) Memory cost: Use of indexing approach will consume a smaller and affordable size of memory space.

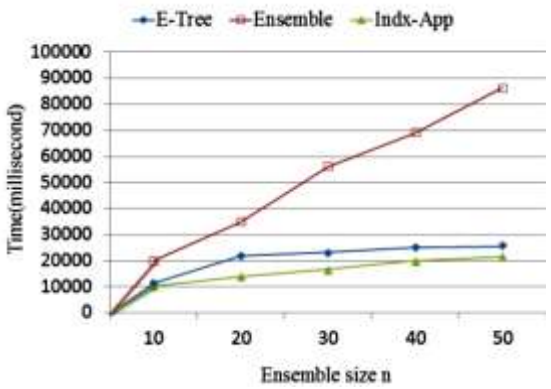

Fig. 2. Comparison with respect to Time

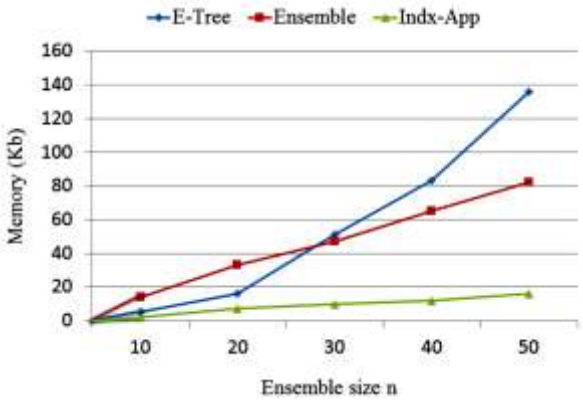

Fig. 3. Comparison with respect to Memory

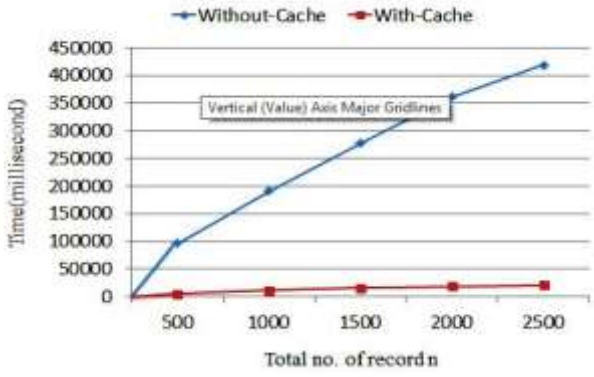

Fig. 4. Comparison of records with caching and without caching

CONCLUSION

In this paper, we addressed several real world problems related to high speed data stream classification. We have provided a solution for the concept drift, concept-evolution and feature evolving problem. Existing data stream classification methods assume that total number of feature in the stream is fixed. Therefore, upcoming instances are misclassified by the existing techniques. We also show how to detect these problems automatically in real time environment. In addition to this, we have proposed the indexing structure that organizes base classifier and helps to achieve lower time complexity for prediction, which is a legitimate research problem well motivated by increasing

real-time applications. The given approach also outperforms previous works in terms classification accuracy, storage capacity and execution speed, while adapting to new trends and patterns in stream data.

REFERENCES

1. C.Aggarwal, "Data Streams: Models and Algorithms," Springer, 2006.
2. Georg Krempl,Indre Zliobaite,Dariusz Brzezinski,Eyke Hullermeier,Mark Last,Vincent Lemaire,Tino Noack,Ammar Shaker,Sonja Sievi,Myra Spiliopoulou,Jerzy Stefanowski , "Open Challenges for Data Stream Mining Research, "SIGKDD Explorations ,Volume 16, Issue 1, pg 1-10,2013.
3. Peng Zhang, Chuan Zhou, Peng Wang, Byron J. Gao, Xingquan Zhu, and Li Guo, "E-Tree: An Efficient Indexing Structure for Ensemble Models on Data Streams," IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING,VOL.27,NO. 2,FEBRUARY 2015.
4. Mohammad M Masud, Tahseen M, Al-khateeb, Latifur Khan, Charu Aggrawal, Jing Gao, Jiawei Han and Bhawani Thuraisinghum ,"Detecting Recurring and Novel classes in Concept Drift Data Streams," IEEE 11th International Conference On Data Mining, pp. 1176- 1181, 2011.
5. Xiao-Li Li, Philip S. Yu, Bing Liu ,See-Kiong Ng, "Positive Unlabeled Learning for Data Stream Classification," SIAM ,pp. 259-270, 2010.
6. P. Zhang, X. Zhu, J. Tan, and L. Guo, "Classifier and Cluster Ensembles for Mining Concept Drifting Data Streams," Proc. IEEE 10th Int'l Conf. Data Mining (ICDM), 2010.
7. H. Wang, W. Fan, P. Yu, and J. Han, "Mining Concept-Drifting Data Streams Using Ensemble Classifiers," Proc. Ninth ACM SIGKDD Int'l Conf. Knowledge Discovery and Data Mining (KDD), 2003.
8. W. Street and Y. Kim, "A Streaming Ensemble Algorithm (SEA) for Large-Scale Classification," Proc. Seventh ACM SIGKDD Int'l Conf. Knowledge Discovery and Data Mining (KDD), 2001.
9. P. Zhang, X. Zhu, Y. Shi, L. Guo, and X. Wu, "Robust Ensembler Learning for Mining Noisy Data Streams", Decision Support Systems, vol. 50, no. 2, pp. 469-479, 2011.
10. A. Bifet, G. Holmes, B. Pfahringer, R. Kirkby, and R. Gavalda,"New Ensemble Methods for Evolving Data Streams, " Proc. 15th ACM SIGKDD Int'l Conf. Knowledge Discovery and Data Mining (KDD), 2009.
11. J. Gao, W. Fan, and J. Han, "On Appropriate Assumptions to Mine Data Streams: Analysis and Practice," Proc. Seventh IEEE Int'l Conf. Data Mining (ICDM), 2007.
12. J. Gao, R. Sebastiao and P. Rodrigues, "Issues in Evaluation of Stream Learning Algorithm,"Proc. 15th ACM SIGKDD int'l Conf. Knowledge Discovery and Data Mining (KDD)2009.
13. Amit Biswas, Dewan Md. Farid and Chowdhary Mofizur Rahman , "A New Decision Tree Learning Approach For novel Class Detection In Concept Drifting Data Stream Classification", journal of computer science and engineering, volume 14, issue 1, july 2012.
14. 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, NO. 5, MAY 2009.
15. Mohammad M. Masud,Qing Chen,Latifur Khan,Charu C. Aggarwal "Classification and Adaptive Novel Class Detection of Feature-Evolving Data Streams," TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, IEEE,pp. 1-14, 2011.
16. G.Divya,D.B.Anand,"An Effective Classification and Novel Class Detection of Data Streams," International Journal Of Engineering And Computer Science ,Vol. 3 Issue 4 ,Page No. 5314-5318, April 2014.
17. P. Zhang,P. Wang, B. Gao and X. Zhu, and Li Guo, "Enabling Fast Prediction for Ensemble Model on Data Streams, " Proc. 17th ACM SIGKDD int'l Conf. Knowledge Discovery and Data Mining (KDD)2011.
18. A. Asuncion and D. Newman, " UCI Machine Learning Repository," http://mlean.ics.uci.edu/database/,2007