Novel Class Detection Using Ensemble Classification Framework

Mathumitha. V
IInd Year – M.E. CSE
Srinivasan Engineering College
Peramabalur
Tamil Nadu, India

Ayyappan. M
IInd Year - M.E. CSE
Srinivasan Engineering College
Peramabalur
Tamil Nadu, India

Jayanthi. S
HOD/CSE
Srinivasan Engineering College
Peramabalur
Tamil Nadu, India

Abstract— **The major problems faced today in the field of Data Stream Mining are infinite length, concept drift, concept and feature evolution. Hence, this project is set to mainly focus on two evolving problems namely concept evolution and feature evolution. First, the Concept evolution occurs when new classes evolve in the huge data set which gives a necessity to find them. Therefore, the novel class detector technique is proposed to identify the novel class and also to enhance the performance in identifying the arrival of novel classes. It is further improved by setting threshold value in the outlier detection mechanism and identifying instances for novel classes. The experimental results prove that the Novel class detection will lead to lowest error rate supported by the new voting method (Hoeffding Option Tree) .In this paper, before examining the data, the classes are usually fixed but when continuously data arrives then not all data are classified. At the same time if some data is misclassified and that particular class is not present in universal existing class then it is known to be a classified as novel class. As soon as the novel class is promised to be detected, then the model is trained which doesn't requires any further classification. This streaming technique will help in the avoidance of traffic to a large extent in network domains and has got huge advantages in money transactions, social networking and other internet allied applications.**

Keywords: Hoeffding option tree,outlier,Novel class

1. INTRODUCTION

A major challenge in data stream classification, which deserves attention but has long assumed that the numbers of classes are fixed. However, in data streams, new classes may often appear. For example, a new kind of intrusion may appear in network traffic, or a new category of text may appear in a social text stream such as Twitter. When a new class emerges, traditional data stream classifiers misclassify the instances of the new class as one of the old classes. In other words, a traditional classifier is bound to misclassify any instance belonging to a new class, because the classifier has not been trained with that class. It is important to be able to proactively detect novel classes in data streams.

For example, in an intrusion detection application, it is important to detect and raise alerts for novel intrusions as early as possible, in order to allow for early remedial action and minimization of damage. A recurring class is a special and more common case of concept-evolution in data streams. It occurs when a class reappears after long disappearance from the stream.

Recurring classes, when unaddressed, create several undesirable effects. First, they increase the false alarm rate because when they reappear, novel class will be falsely identified, whereas such classes may observe normal representative behavior. Second, they also increase human effort, in cases where the output of the classification is used by human analyst. In such cases, the analyst may have to spend extra effort in analyzing the afore-mentioned false alarms. Finally, "novel class detection" has additional computational effort, which is costlier than regular "classification" process. Novel class detection is major concept of concept evolution. In data stream classification assume that total no of classes is fixed but not be valid in a real streaming environment. When new class may evolve at any time.

The goal of the project is designed to function as a multi-class classifier for concept-drifting data streams, detect novel classes, and distinguish recurring classes from novel classes. Keep an ensemble of size, and also keep an auxiliary ensemble where at most models per class are stored. This auxiliary ensemble stores the classes in the form of classification models even after they disappear from the stream. Therefore, when a recurring class appears, it is detected by the auxiliary ensemble as recurrent.

This approach greatly reduces false alarm rate as well as the overall error. If, however, a completely new class appears in the stream, it is detected as novel by the auxiliary ensemble as well. This is the first work that addresses the recurring concept-evolution in data streams and class issue.

Proposed solution, which uses an auxiliary ensemble for recurring class detection, reduces overall classification error and false alarm rates. Second, this technique can be applied to detect periodic classes, such as classes that appear weekly, monthly, or yearly. It will be useful for better predicting and profiling the characteristics of a data stream. Finally, apply our technique on a number of real and synthetic datasets, and obtain superior performance over state-of-the-art techniques.

2. RELATED WORK

In an existing, concept evolution and feature evolution problem was ignored.the incremental learning approaches is to handle the infinite length and concept drift. A single model incremental approach,where a single model is maintained with a new data.the single model incremental approach is overcome by hybrid batch incremental approach.the hybrid batch incremental approach will require simpler operations to update model.but,it will ignore concept evolution and feature evolution problem.

In cluster based technique it will handle concept evolution,but it is not applicable to multiclass data stream classification.Mohammad et all.. it has data stream classification problem under dynamic sets.It has a multiclass classifier and novel class detector by ensemble of models to classify unlabelled data.but,still it has a feature evolution problem.It address all the challenges .but,it does not distinguish the actual arrival of a novel class.Shruti et all. Hoeffding bound will classify the data accurately. The concept evolution and feature problem was ignored. Masud et al.Address the flexible and dynamic adaptive decision boundary for outlier detection and also distinguish more than one novel class. However,feature evolution is not addressed.

In existing system use act miner applies an ensemble classification technique but used for limited labeled data problem and addressing the other three problem so reducing the cost. Act miner is extends from mine class. Act miner integrates with four major problem concept drift, concept evolution, novel class detection, limited labeled data instances. But in this technique dynamic feature set problem and multi label classification in data stream classification. Based on clustering methods for collecting potential novel instances so memory is required to store. Another disadvantage is that using clustering method first find centroid. And also incremental so time overhead occurs. And also not possible classify streamed data continuously. Because streamed data continuously come and classification become continuous task.

Mining streaming data is one of the recent challenges in data mining. Data streams are characterized by a large amount of data arriving at rapid rate and require efficient processing. Moreover, the data may come from non-stationary sources, where underlying data distribution changes over time. It causes modifications in the target concept definition, which is known as concept drift. The main types of changes are usually divided into sudden or gradual concept drifts depending on the rate of changes. Classical static classifiers are incapable of adapting to concept drifts, because they were learned on the out-of-date examples. This is the reason why their predictions become less accurate with time. Some methods have already been proposed to deal with the concept drift problem.

They can be divided into two main groups: trigger based and evolving. Trigger-based methods use a change detector to identify the occurrence of a change. If the change is detected, then the online classifier, connected with the detector, is re-trained. One of the most popular detectors is DDM described. On the other hand, evolving methods attempt to update their knowledge without explicit information whether the change occurred. An example of such methods is an adaptive ensemble.

This paper focuses mainly on block-based ensembles, which component classifiers are constructed on blocks (chunks) of training data. In general, a block-based approach operates in a way that when a new block is available, it is used for evaluation of already existing component and for creation of a new classifier. The new component usually replaces the worst one in the ensemble.

3. OUR SYSTEM AND ASSUMPTIONS

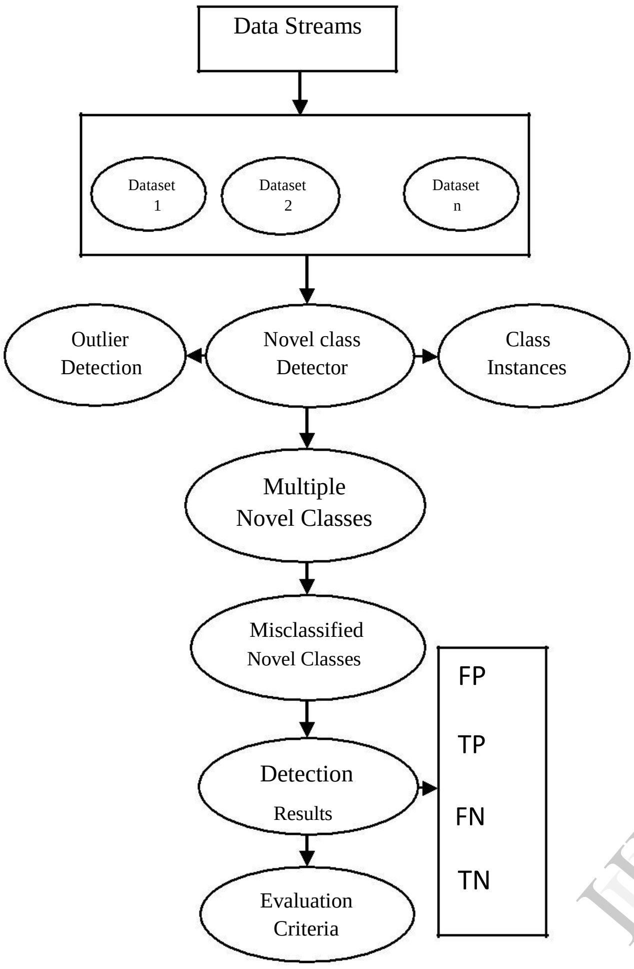
Data Stream means continuous flow of data. Examples of data stream are computer network traffic, phone conversation, Web Searches, Sensor data and ATM transaction. Data Stream Mining is a process of extracting knowledge from continuous, rapid data records. It can be considered as a subfield of data mining. Data Stream can be classified into offline streams and online streams.

Online Data stream mining used in a number of real world applications, such as intrusion detection, network traffic monitoring and credit card fraud detection. And offline data stream mining used in like generating report based on web log streams. Characteristics of data stream are continuous flow of data. Data size is extremely large and potentially infinite. It's not possible to store all data. Data stream classification three major problems occurred.

1. Infinite Training Data Can't store or use all historical data for training.
2. Concept drift: Data changes over time. Historical training data built a model on those data which are outdated.
3. Novel class: Novel class may appear over time. Old classes become obsolete (out dated).

Data stream have infinite length multi pass learning algorithm can not applicable as they would required infinite storage. Concept drift occurs when data changes over time. Another major problem is ignored by state of art data stream classification techniques which is concept evolution that means emergence of novel class. Assume that total no of classes is fixed. But in real data stream classification problems such as text classification and fault detection Novel class may appear at any time in a stream. So all novel class instance go undetected until novel class manually detected by experts.

Novel class detection is major concept of concept evolution. In data stream classification assume that total no of classes is fixed but not be valid in a real streaming environment. In this project proposed novel class detector to analyze the novel classes. Then use the outlier detection using adaptive threshold. Perform the novel class detection using Gini coefficient, and identify the simultaneous multiple novel class detection.



4. SYSTEM PRELIMINARIES

NOVEL CLASS DETECTION:

Algorithm 1. Detect-Novel(M,Buf)

Input:

M: Current ensemble of best L classifier.

Buf: Buffer temporarily holding F-outlier instances

Output: The novel class instances identified, if found

```
1.  K0 ← (k* |Buf|/S) // S= chunk size k= cluster per chunk
2.  H ← k-means(Buf,k0)// create k0 o-pseudopoints
3.  for each classifier Mi € M do
4.     tp ← 0
5.     for each cluster h€H do
6.        h.sc ← q-NSC(h)
7.        if h.sc > 0 then
8.           tp+= h.size // total instances in the cluster
9.           for each instances x € h.cluster do x.sc ← max(x.sc,h.sc)
10.       end if
11.    end for
12.    if tp>q then vote ++
13. end for
14. if vote == L then // found novel class, identify novel instances
15.    X nov ← all instance x with x.sc > 0
16.    for all x € X nov do
17.       x.ns > Nscore(x)
18.       if x.ns > Gini th then N_list ← N_list U x
19.    end for
20.    Detect-Multinovel(N_list)
21. end if
```

The input to the algorithm is the ensemble M and the buffer Buf holding the outliers instances. At first, we create K0 clusters using K-means with the instances in Buf (line 2), where K0 is proportional to K, the number of pseudo points per chunk (line 1). Then each cluster is transformed into a pseudo point data structure, which stores the centroid, weight (number of data points in the cluster) and radius (distance between the centroid and the farthest data point in the cluster). Clustering is mainly for speed up the computation of q-NSC value. If we compute q-NSC value for every F-outlier separately, it takes quadratic time in the number of the outliers. On the other hand, if we compute the q-NSC value of the K0F-outlier pseudo points it takes constant time.

DECISION BOUNDARY OUTLIER

Detection:

The novel class detection process contains three steps. First, a decision boundary is built on the period of training. Second, test points falling outside the decision boundary are named as outliers. Finally, the outliers are analyzed to see if there is enough cohesion among themselves (i.e., among the outliers) and separation from the existing class instances. Decision boundary for outlier detection by allowing a slack space outside the decision boundary.

The threshold value will control the space, and the threshold value is adapted continuously to reduce the false alarms and missed novel classes. Second, apply a probabilistic approach to detect novel class instances by discrete Gini Coefficient. With this approach, able to distinguish different causes for the appearance of the outliers. Derive an analytical threshold for the Gini Coefficient that identifies the case where a novel class appears in the stream.

5. CONCLUSION

The novel class detection is the major problem in data streams. Existing novelty detection techniques either assume that there is no concept-drift, or build a model for a single “normal” class and consider allother classes as novel. In this project, proposed the Novel class detector and analyze the misclassified novel class. Use the Novel class detector to enhance the novel class detection module by making it more adaptive to the evolving stream, and enabling it to detect multiple novel classes at a time. In this paper, introduce new voting method to detect novel class using Hoeffding Option Tree in concept drifting data stream classification which builds a decision tree from data stream. Here models are trained when potential novel instance is found and not require to collect misclassified instances. So do not require further classification. Time and accuracy is improved.

REFERENCES

[1]. C.C. Aggarwal, “On Classification and Segmentation of Massive Audio Data Streams,” Knowledge and Information System, vol. 20, pp. 137-156, July 2009.

[2]. C.C. Aggarwal, J. Han, J. Wang, and P.S. Yu, “A Framework for On-Demand Classification of Evolving Data Streams,” IEEE Trans. Knowledge and Data Eng., vol. 18, no. 5, pp. 577-589, May 2006.

[3]. A. Bifet, G. Holmes, B. Pfahringer, R. Kirkby, and R. Gavalda`, “New Ensemble Methods for Evolving Data Streams,” Proc. ACM SIGKDD 15th Int’l Conf. Knowledge Discovery and Data Mining, pp. 139-148, 2009.

[4]. S. Chen, H. Wang, S. Zhou, and P. Yu, “Stop Chasing Trends: Discovering High Order Models in Evolving Data,” Proc. IEEE 24th Int’l Conf. Data Eng. (ICDE), pp. 923-932, 2008.

[5]. W. Fan, “Systematic Data Selection to Mine Concept-Drifting Data Streams,” Proc. ACM SIGKDD 10th Int’l Conf. Knowledge Discovery and Data Mining, pp. 128-137, 2004.

[6]. J. Gao, W. Fan, and J. Han, “On Appropriate Assumptions to Mine Data Streams,” Proc. IEEE Seventh Int’l Conf. Data Mining (ICDM), pp. 143-152, 2007.

[7]. S. Hashemi, Y. Yang, Z. Mirzamomen, and M. Kangavari, “Adapted One-versus-All Decision Trees for Data Stream Classification,” IEEE Trans. Knowledge and Data Eng., vol. 21, no. 5, pp. 624-637, May 2009.

[8]. G. Hulten, L. Spencer, and P. Domingos, “Mining Time-Changing Data Streams,” Proc. ACM SIGKDD Seventh Int’l Conf. Knowledge Discovery and Data Mining, pp. 97-106, 2001.

[9]. I. Katakis, G. Tsoumakas, and I. Vlahavas, “Dynamic Feature Space and Incremental Feature Selection for the Classification of Textual Data Streams,” Proc. Int’l Workshop Knowledge Discovery from Data Streams (ECML/PKDD), pp. 102-116, 2006.

[10]. I. Katakis, G. Tsoumakas, and I. Vlahavas, “Tracking Recurring Contexts Using Ensemble Classifiers: An Application to Email Filtering,” Knowledge and Information Systems, vol. 22, pp. 371-391, 2010.