Neurocomputing 158 (2015) 234–245

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

Novel class detection in data streams using local patterns and neighborhood graph

Poorya ZareMoodi, Hamid Beigy *[,1], Sajjad Kamali Siahroudi

Department of Computer Engineering, Sharif University of Technology, Azadi Avenue, Tehran, Iran

ARTICLE INFO

Article history:
Received 21 July 2014
Received in revised form
24 October 2014
Accepted 17 January 2015
Communicated by: A.M. Alimi
Available online 7 February 2015

Keywords:
Novel class detection
Data stream
Concept drift
Classification

ABSTRACT

Data stream classification is one of the most challenging areas in the machine learning. In this paper, we focus on three major challenges namely infinite length, concept-drift and concept-evolution. Infinite length causes the inability to store all instances. Concept-drift is the change in the underlying concept and occurs in almost every data stream. Concept-evolution, in fact, is the arrival of novel classes and is an undeniable phenomenon in most real world data streams. There are lots of researches about data stream classification, but most of them focus on the first two challenges and ignore the last one. In this paper, we propose new method based on ensembles whose classifiers use local patterns to enhance the accuracy. Local pattern is a group of Boolean features which have local influence on ordinal and categorical features. Also, in order to enhance the accuracy of novel class detection we construct a neighborhood graph among novel class candidates and analyze connected components of the constructed graph. Experiments on both real and synthetic benchmark data sets show the superiority of the proposed method over the related state-of-the-art techniques.

1. Introduction

The purpose of data stream classification is to determine which category an observation belongs to. These observations are part of an infinite length data stream. Infinite lengths, high speed, limitation of response time and concept drift are challenges that we face in classifying data streams. There are many researches that address the aforementioned challenges [1–9]; however, most of them ignore another major challenge "concept evolution" which led to the emergence of novel classes.

In the most real world data streams emergence of novel classes is an inevitable phenomenon. For example, a new kind of intrusion may appear in network traffic, or a new category of stones may be discovered by a Mars rover. Therefore, instances from all classes are not available at the start of the stream in order to train a learner. Also, the exact number of classes is unknown at first. In such a case, the goal of the learner is to accurately classify the instances that belong to existing classes and simultaneously detect emergence of novel classes. In this case, an existing class is defined as a class where at least one of its instances has been observed from the start of the stream. The general workflow and conditions for novel class detection in data streams are given below.

In the initial phase of training, learner uses the first M instances of the stream as the training set to build an initial classifier model (M is usually a small number). After building the initial classifier, for each new arrival instance, the learner has to determine the instance that belongs to either one of the existing classes (and specifically, which one) or a novel class. In order to detect the emergence of a novel class, the learner should see a group of newly arrived instances, not just one. Therefore, classification can be postponed until enough instances are seen by the learner to gain confidence in deciding whether the instance belongs to a novel class or not. However, there is a maximum allowable time up to which the learner can postpone classification of each instance. In supervised methods the learner receives true label of each instance within a time limit (or immediately) after the instance is classified. The learner updates its model periodically with respect to the observed instances.

There are two conditions that must be verified to declare the emergence of a novel class: cohesion–separation condition and threshold condition. The former condition is based on the cluster assumption and implies that the novel class instances must be more similar to each other than being similar to instances of other classes. The latter condition implies that the number of the candidate instances for a novel class must be more than a given threshold q. The threshold is being used to distinguish between

* Corresponding author. Tel.: +98 21 6616 6624; fax: +98 21 6601 9246.
E-mail addresses: pzare@ce.sharif.edu (P. ZareMoodi), beigy@sharif.edu (H. Beigy), skamali@ce.sharif.edu (S. Kamali Siahroudi).
[1] URL:http://sharif.edu/~beigy/

http://dx.doi.org/10.1016/j.neucom.2015.01.037
0925-2312/

outliers and novel class instances; when the number of candidates is less than the given threshold, we assume that they are outliers of existing classes. The appropriate value of threshold is determined by experts and is based on application.

Due to aforementioned conditions, a leaner has to check a group of instances in order to detect emergence of a novel class. Therefore, existing methods are either chunk-based [10–12] or based on time constraints [13,14]. Also, these methods can be divided into two learning categories: supervised [13,10,11] and unsupervised [15,14]. In the next section, we briefly discuss these categories.

A new supervised chunk-based approach for joint novel class detection and classification problem is proposed in this paper. The proposed method utilizes local patterns, which are based on the impact of some categorical features on the range of values for ordinal and continuous features. In addition, our method constructs a graph using novel class candidates and analyze its connected components. Using graphs help to check cohesion and separation more accurately. Like almost all existing approaches, we use ensemble learners however, we define new measures in order to update ensemble that enhances our method's precision. We call the proposed algorithm as LOCE (LOcal Classifier Ensemble). We apply LOCE on a number of real and synthetic benchmark data sets, and obtain superior performance over the state-of-the-art methods.

The rest of this paper is organized as follows. The related work discussed in Section 2. The proposed method is given in Section 3. Experimental results are presented in Section 4. This paper concludes with conclusions and future works in Section 5.

2. Related work

Novelty detection [16–19] is similar to novel class detection. In novelty detection, observations which are non-similar to existing classes considered as novelty. However in novel class detection there are more conditions to be verified. First of all, similarity between novel class candidates have to be more than the similarity between them and the existing classes. Second, the number of candidates has to be more than a given threshold.

Existing approaches for novel class detection, based on their input data can be divided into supervised and unsupervised categories. As mentioned before, learners use the first M instances of the stream to build an initial model. In unsupervised methods, learner only receives the true labels of these instances and does not receive the true labels of subsequent instances. Methods that fall within this category, update their model based on predicted labels. On the other side, supervised methods expect to receive the true label of an instance after predicting its label. Therefore, updating in this category is based on true labels. Our proposed method also belongs to this category. Comparison between these two categories is shown in Table 1. In what follows, we briefly discuss well-known existing works of these methods.

1. *Unsupervised methods*: To the best of our knowledge, only two unsupervised methods have been proposed for novel class detection in data streams. First approach for detecting novel classes in data streams is OLINDDA [15]. It addresses the problem of novel class detection as an extension of one class classification. The goal of this method is to get a training set as normal concept and then classify instances that belong to this concept as “normal” and instances of new concepts as “novel”. In order to reach the aforementioned goal it builds K clusters with the training data. The normal model is composed of K hyperspheres obtained directly from these clusters. The center of each hypersphere is the centroid of its cluster, and its radius is the Euclidean distance from the centroid to the farthest example of the respective cluster. Each new (unseen) instance if located within the hyperspheres, then it will be classified as “normal” and otherwise it will be a candidate for a new concept. In order to distinguish between outliers of normal concept and novel concepts, it builds some clusters with candidates periodically. These candidate clusters are then evaluated in an attempt to find probable new concepts (by verifying cohesion–separation and threshold conditions).

OLINDDA considered training data as a whole concept and if they are from more than one concept, then it cannot distinguish between these concepts. MINAS [14] is a multi-class version of OLINDDA. In order to enable learner to distinguish between normal concepts, the authors of MINAS proposed to keep a set of hyperspheres for each observed concept in the initial training phase. The steps of this approach are similar to the OLINDDA's steps.

2. *Supervised methods*: ExMiner [13] is one of the earliest and well-known supervised methods for novel class detection in data streams. It uses ensemble of multi-class classifiers. For each classifier, it determines a decision boundary which defines the physical boundary of the training data which used to train the classifier. Union of the decision boundaries of all classifiers shows the physical boundary of existing instances. Instances located outside of the boundaries are candidates for novel classes. Cohesion among novel class candidates checked periodically in order to detect emergence of novel classes. This method has an issue which can lead to forget some classes in the updating procedure. Consider a scenario in which the number of classifiers in the ensemble has reached its limit and learner also wants to add a new classifier to the ensemble. Authors of ExMiner proposed to discard the classifier with the highest error in such a situation. However, ExMiner uses multi-class classifiers and therefore it is possible that a classifier has the lowest accuracy but it is the only one which can detect a specific class. In such a case ExMiner forgets the specific class and cannot detect it anymore.

CLAM [10] is proposed to address the forgetting class issue in ExMiner. It is similar to ExMiner but keeps an ensemble of one-class classifiers per each observed class. Therefore, it keeps at least one classifier per each observed class. Union of the decision boundaries of all classifiers shows the physical boundary of existing instances. The steps of this approach are similar to the ExMiner's steps. This method does not forget any of observed classes. However, sometimes underlying distribution of data is changed and it is necessary to forget some classes. Hence, the remaining classifiers of these classes can have negative impact on classification accuracy.

Both ExMiner and CLAM methods compute q-NSC (q-Neighborhood Silhouette Coefficient) for each novel class candidates. q-NSC is a unified measure of cohesion and separation, whose value lies between the range $[-1,+1]$. A positive q-NSC means that the instance is closer to the other candidates (more cohesion) and farther away from existing class instances (more separation) and vice versa. After this, if the number of instances with positive q-NSC is more than a given threshold then emergence of novel class will be declared. The drawback is that similarity between positive q-NSC instances will not be checked and thus wrong declaration of novel class emergence is possible. Assume the scenario in which two or more novel classes emerge concurrently. Also, the number of instances from each novel class is less than the threshold but

Table 1
Comparison between supervised and unsupervised methods for novel class detection in data streams.

Feature	Supervised	Unsupervised
Access to the true label of instances 1 to M	✓	✓
Access to the true label of instances after M	✓	×
Update continuously with data stream progression	✓	✓
Update the model based on predicted labels	×	✓
Update the model based on true labels	✓	×