ATLANTIS PRESS
Advances in Computer Science Research, volume 74
2nd International Conference on Computer Engineering, Information Science & Application Technology (ICCIA 2017)

A Novel Sampling Strategy for Active Learning over Evolving Stream Data

Xuxu Zhang [a], Zhi Cao [b], Li Peng [c] and Siqi Ren [d]

College of Computer Science & Electronic Engineering, Hunan University, Changsha 410000, China

[a]zhangxuxu96379@163.com, [b]66384436@qq.com, [c]6521982@qq.com, [d]siqirenzl@163.com

Keywords: Active learning, Data streams, Evidence, random strategy.

Abstract. In classification tasks, data labeling is an expensive and time-consuming process, hence, active learning which query labels for a small representative portion of data, is becoming increasingly important. However, few works consider the challenges from data steam setting because most of the active learning method is designed for non-streaming setting. Be based upon the status quo, after synthesizing the evidence-based uncertainty sampling strategy and split sampling strategy above, we propose a new sampling strategy for active learning over evolving stream data, which can take full advantages of the strengths of each. First, the original data stream is randomly divided into two sub-streams. Instances from one sub-stream are labeled according to the high evidence-focused uncertainty strategy, while instances from the other sub-stream are marked by the random strategy for detecting true concept drifts. Second, we introduce a sliding window in the high evidence-focused uncertainty strategy, finding out whether an instance is the conflict-uncertainty instance or not. Clearly, our strategy solves the issue of the effective use of evidence in data streams setting, and can choose more representative instances over evolving data streams for training a model. Finally, in experiments over four benchmark datasets, compared with state-of-art active learning strategies, the result illustrates good predictive performance of our proposed approach.

1. Introduction

Nowadays, more and more data are being generated continuously by networks, such as sensor networks, social networks, web applications and financial activities etc. Unlike traditional datasets, data items within a data stream are temporally ordered, fast-changing, generally large-scale, and potentially infinite [1].

For learning predictive models on data stream, it is possible to access the true labels of the instances continuously. Unfortunately, inherent labeled instances in data streams are very scarce in practice. Conversely, a very limited number of labeled instances can be collected, and they can hardly provide enough information to train models with good generalization capabilities [2]. However, the manual labeling is expensive, especially in terms of time. Besides, in pace with time, the relationship between attributes and labels might change, such as spam identification and vaccine production. In order to know the true label, it is needed to scan the mail or make a laboratory test, which is time-consuming. Hence, querying labels for a small representative subset of all stream data, has become an effective solution. Such a learning situation goes by the name of active learning. In pool-based and online environments [3, 4], active learning has received widely attention and research.

In the data stream setting, active learning is further divided into online active learning and active learning in data streams. The main difference between the two branches is whether the concept drifts exist or not. Online active learning has a generally accepted assumption that the relationship between attributes and labels is stationary [5]. So, the strategies relate to online and stationary setting, mainly select instances near the decision boundary and may never label instances from remote regions. Inversely, in evolving data streams, the mapping relation between the input data and the label may alter over time in the instance space. For capturing concept drifts as they happen, it is important to select some instances from remote area or preserve the incoming data distribution [6].

In the pool-based environment, [7] suggests that we should consider two type of uncertainties, namely, conflicting-evidence uncertainty and insufficient-evidence uncertainty. Unfortunately, in the active learning on data streams, there is no works that employ the evidence to choose more representative instances. Motivated by this tangible, we propose a novel strategy that integrates the evidence into split strategy. Instances from one sub-stream are labeled in the light of the high evidence-focused uncertainty strategy, while instances from the other sub-stream are marked by the random strategy for detecting true concept drifts.

The remainder of this paper is organized as follows. The next section mainly overviews related work in the area of data stream and active learning. Our proposed method is described in details in the third section. In Section four, we present experimental results for a number of real world datasets and a sensitivity analysis of the sliding window size is provided. In the last section, we conclude this study and look forward to the future.

2. Related Work

We briefly review concept drift in data streams, and the most advanced active learning strategy in data streams. In addition, we introduce the definition and character of evidence.

2.1 Concept Drift

Active learning with static data usually employs some heuristics or rules in selecting the most valuable instances, such as uncertainty sampling, query by committee [8], and query by margin [9]. Comparing to the traditional active learning, active learning over evolving data streams has to face a challenge, named "concept drift". In a general, two main types of concept drifts are distinguished: real concept drift and virtual concept drift. The former type refers to changes in $P(y|x)$, where virtual concept drift happens only when the distribution $P(x)$ of the incoming data changes. In other words, only when the class conditional distribution changes can we make a conclusion that the true concept drift happens. For the purpose of dealing with concept drift, several drift detection methods [10, 11] and active learning strategies [5, 6] have been proposed for data streams.

2.2 Active Learning Strategy

The Random strategy is naive and simple. It only uses a parameter β that is equal to the pre-defined budget B. The uncertainty strategy is a frequently used method in a static setting that chooses the instances about which the current classifier is the least certain. Obviously, uncertainty strategy pays close attention to the instances lied in classification boundary region, and scarcely selects instances far from the boundary.

In data stream setting, to solve the challenge: how to effectively find and handle the concept drift in the context of maintaining the accuracy of classification, some works use randomization to catch possible concept drift, such as selective sampling, uncertainty strategy with randomization (RU), DBALStream [5]. Recently, to overcome the problem that change detectors may not distinguish virtual concept drift from true concept drift, split strategy [6] was introduced to split a stream into two sub-streams at random with pre-defined probability v. Clearly, researching on active learning over evolving data stream is comparably new, hence, only a few active learning methods are designed for it.

In 2016, Sharma et al. further divided the traditional uncertainty into two categories according to the reason of instance uncertainty, namely conflicting-evidence uncertainty (UNC-CE) and insufficient-evidence uncertainty (UCN-IE) [7]. The experiments also showed that the conflicting uncertain instance has a more obvious effect on the improvement of classification performance.

3. Our Method

In this section, we describe our novel sampling strategy for active learning over evolving data streams, as showed in Algorithm 1. Once the arrival of a new instance, we will process it and make a decision on whether or not to perform manual marking.

Briefly speaking, our strategy, firstly, divide original data stream into two parts and secondly, one part uses the random strategy while the other part utilizes the high evidence-focused uncertainty

strategy. Obviously, it not only uses the random strategy for preserving the incoming data distribution and capturing the true concept drift, but also considers the type of uncertainty, namely, conflicting-evidence uncertainty and insufficient-evidence uncertainty. By using this strategy, we can consume less labeling cost and training time to improve classification accuracy.

Algorithm 1 Esplit(x_i, L, v, B)

Input: x_t: an new incoming instance, L: trained classifier L, v ∈ $(0, 1)$: proportion of random labeling for capturing true drift, B: labeling budget, b_t: labeling cost of labeled instances at time t.

Output: labeling ∈ $\{true, false\}$

```
if (b_t < B) then
    generate a uniform random variable η ∈ (0, 1);
    // that is η~U[0, 1]
    if (η < v) then
        return labeling = RandomStrategy(B);
    else
        return labeling = EUncertaintyStrategy( x_t, L, s, W );
    end if
    updating b_t
end if
```

Fig. 1 A novel active learning strategy

In the high evidence-focused uncertainty strategy which is depicted in Algorithm 2, we introduce a buffer and a dynamic threshold value, respectively namely evidences and θ_e. The evidences work as a queue data structure (First in And First Out, FIFO): the oldest one in the queue is deleted and the new one is push at the end. When concept drift is detected, we will clear evidences. Firstly, we use margin-based metric to measure the uncertainty of an instance. The margin for instance x_i is defined as:

$$\text{margin}(x_i) = P_L(y_m|x_i) - P_L(y_n|x_i) \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots (1)$$

The evidence of an uncertain instance is estimated and used in a batch scenario or pool setting in [7]. However, in the data stream setting, it is impossible to store all instances in the whole data stream and calculate the evidence of all unlabeled instances for ranking, and then pick some unlabeled instances that have higher evidence and lower confidence than the other instances. In addition, as uncertainty threshold changes over time in data stream, we will not directly take advantage of the calculation result of [7] and will process it according to the following formula:

$$E_{Margin}(x_i) = \text{E}(x_i)/margin(x_i) \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots (2)$$

Where $\text{E}(x_i)$ represents the calculation result in [7]. Clearly, $E_{Margin}(x_i)$ is inversely proportional to $margin(x_i)$ and is proportional to $\text{E}(x_i)$.

Next, after calculating$E_{Margin}(x_i)$, we need to calculate the ranking in evidences of it according to the following formulas:

$$SL(x_i) = \sum_{evidences_j \in evidences} \mathbb{I}\{evidences_j > E_{Margin}(x_i)\} \cdots\cdots\cdots (3)$$

$$\text{DL}(x_i) = SL(x_i)/(|evidences| + 1) \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots (4)$$

Where $\mathbb{I}$ is an indicator function that returns 1 if the condition is true, and 0 otherwise. $evidences_j$ is the i-th element in evidences. $|evidences|$ is the size of evidences. $SL(x_i)$ and $\text{DL}(x_i)$ respectively represent static ranking and relative ranking of $E_{Margin}(x_i)$ in evidences.

Finally, we need to determinate the type of uncertainty. Due to the dynamic nature of data streams, a dynamic threshold θ_e is used to verity if a new incoming uncertain instance is conflicting-evidence.

$$isConflicting(x_i) = \mathbb{I}(\text{DL}(x_i) < \theta_e) \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots (5)$$

Where θ_e is dynamically adjusted. When the relative ranking $\text{DL}(x_i)$ is under the thresholdθ_e, we will approximately deem that this instance is conflicting-evidence uncertain.

Algorithm 2 EUncertaintyStrategy(x_t, L, s, W)

Input: x_i: an new incoming instance, L: trained classifier L, θ_e: the threshold to see if an incoming uncertain instance is conflicting-evidence, s ∈ $(0, 1]$: Threshold adjustment step, W: size of evidences of previous uncertain instances, $Evidences$: an evidences array
Output: labeling ∈ $\{true, false\}$
Initialize: θ ← 1, θ_e ← 1, s ← 0.01

```
margin(x_i) = P_L(y_m|x_i) − P_L(y_n|x_i)
if (margin(x_i) < θ) then     //the instance is uncertain
    θ = θ ∗ (1 − s);
    calculate E( x_i )
    according to formula (2), calculate E_Margin(x_i)
    according to formula (3), (4), calculate DL(x_i), and push E_Margin(x_i) into evidences
    according to formula (5), calculate isConflicting(x_i)
    if ( isConflicting(x_i) = true) then
        //is conflicting-evidence uncertain
                θ_e = θ_e ∗ (1 − s);
                return labeling = true;
            else
                θ_e = θ_e ∗ (1 + s);
                return labeling = false;
            end if
        else
            θ = θ ∗ (1 + s);
            return labeling = false;
        end if
```

Fig. 2 High evidence-focused uncertainty strategy

4. Experiments

4.1 Datasets and Experimental Settings

Four real world public datasets are introduced in our experiment, named Forest Covertype, Poker-Hand, Electricity, and Airlines.

Forest Cover type dataset is a collection of description information of seven forest cover types of the geographical space, and is often seen as benchmark for evaluating data stream classifier. The goal is to predict forest cover types from cartographic variables. Each record in the Poker-Hand dataset is made up of five cards in standard poker (the total of 52 cards). The Electricity dataset can be used to make a prediction of change trend of electricity price in New South Wales in Australia. Airlines dataset collects raw flight schedule information from US flight control and is used to predict whether a given flight will be delayed or not. The more characteristics are showed in TABLE 1.

Table 1 Dataset Characteristics

Dataset name	Number of Instances	Number of Attributes	Number of Classes
Forest Covertype	581012	54	7
Poker-Hand	829201	11	10
Electricity	45312	8	2
Airlines	539383	7	2

All the experiments are performed in Massive Online Analysis (MOA) platform [12]. In order to ensure fairness, all five algorithms use Single Classifier Drift (SCD) as base classifier. For all methods, the first 500 instances of each dataset are used to train the initial models, and the rest are leveraged to perform the evaluation. SCD is configured by early drift detection method (EDDM) [13], which employs the distance-error-rate to detect concept drift. The other parameters in experiments are: W=40, θ=1, θ_e=1, s=0.01, v=0.5.

Besides, we compare Espilt with the four active learning strategy, including random strategy, variable uncertainty strategy with randomization (RanVarUn), and split strategy (Split).

4.2 Accuracy Evaluation

In the four real dataset, we compare our method with other algorithms under different manual marking ratio, as shown in figure 3.

The result shows that the overall classification accuracy of Esplit algorithm is higher than the other method, especially in the Airlines and Electricity datasets. In the Forest Covertype dataset and Poker-Hand dataset, the Esplit algorithm is superior to the other algorithms when the proportion of artificial markers is 10%-20%. In the actual application process, the proportion of artificial labeling is about 15%, so our proposed Esplit algorithm has great practical significance and application value.

4.3 Influence of the Parameter W

In our Esplit algorithm, a queue data structure is introduced. It is necessary to set a maximum length before taking advantage of it. The experiment result is showed in figure 4.

Through the observation, it is found that the parameter W has very little influence on the classification result. On the Forest Covertype dataset and Airlines dataset, when the value of parameter W is 50, they have the best overall classification performance. On the Poker-Hand dataset and Electricity dataset, when W is 30, our method can provide more stable and effective improvement on classification performance. Besides, we also observes that when the value of W is greater than 50, the classification performance is not better, because historical information is too much to provide correct guidance in the process of adjusting of the type of uncertainty.In a word, when the window size is between 30 and 50, the algorithm has a better performance, and it can make a good judgment on the type of uncertain instances in the data stream.

5. Conclusion

Under the constraints of labeling cost of data streams, learning classification models through limited and labeled instances is becoming more and more ordinary.

In this paper, our approach combines split strategy and evidence in order to improve sampling. When a new incoming instance is uncertain, we further determine whether the instance is conflicting-evidence uncertain, and label this instance only if the judgment is true. On four real-world datasets, experimental results demonstrated that our approach outperforms most advanced active learning strategies in terms of predictive accuracy. The results also showed that the parameter W only have very little impact the predictive accuracy. When the window size is between 30 and 50, the algorithm has a better performance, and it can make a good judgment on the type of uncertain instances in the data stream.

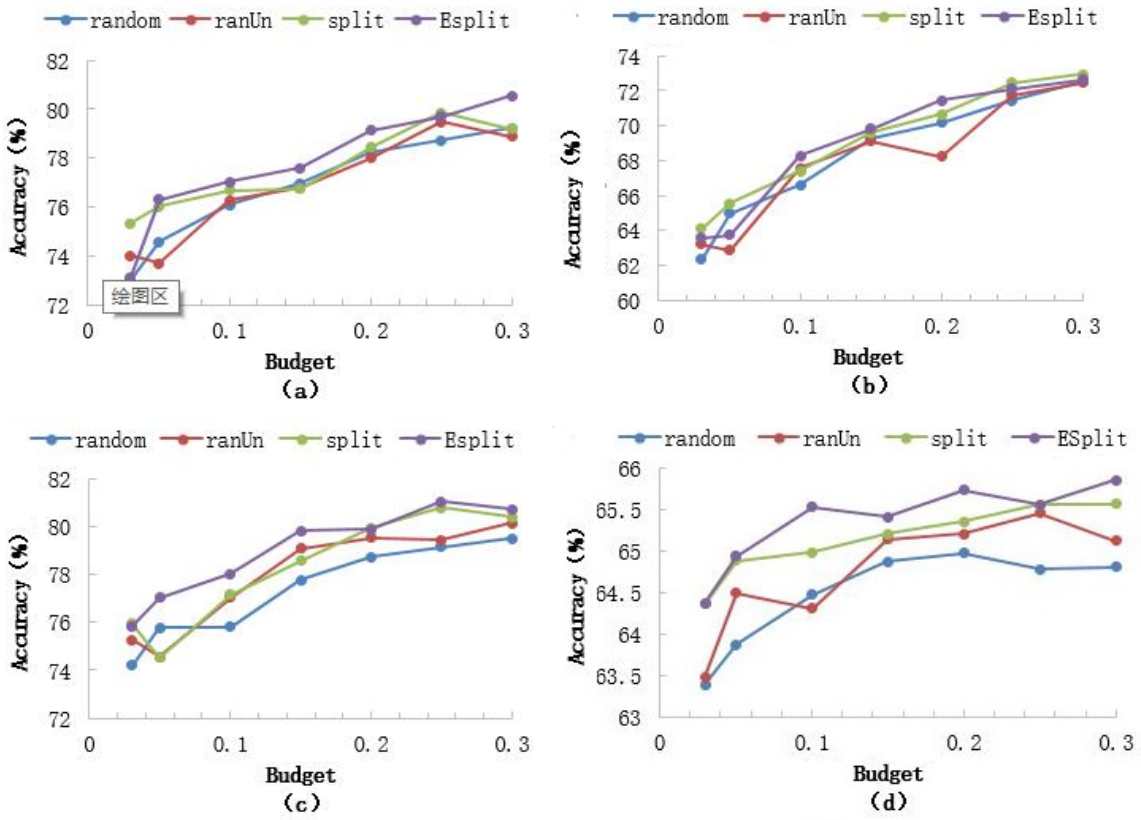

Fig. 3 Accuracy on: a) Forest Covertype b) Poker-Hand c) Electricity and d) Airlines

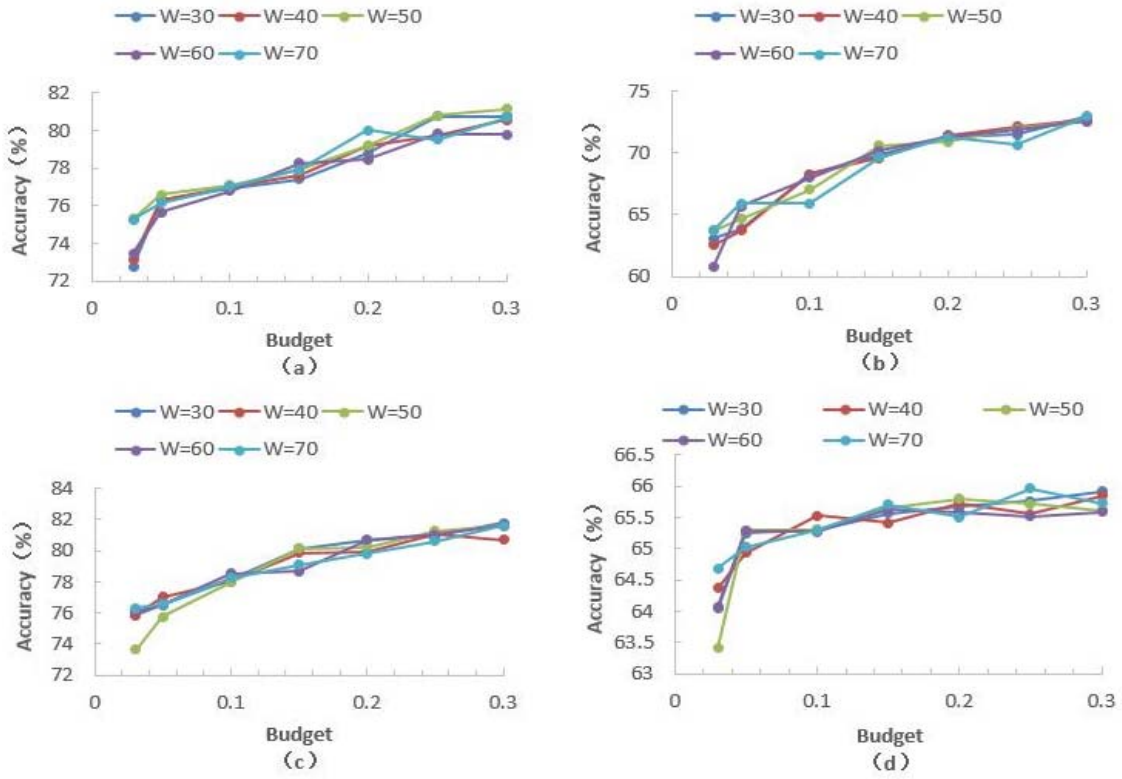

Fig. 4 Accuracy on: a) Forest Covertype b) Poker-Hand c) Electricity and d) Airlines varying the evidences size from 30 to 70 with step 10

References

[1]. Domingos P, Hulten G. Mining high-speed data streams[C]// ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM, 2002:71-80.

[2]. Zhu X, Zhang P, Lin X, et al. Active Learning from Data Streams[C]// IEEE International Conference on Data Mining. IEEE, 2007:757-762.
[3]. Goudjil M, Koudil M, Bedda M, et al. A novel active learning method using SVM for text classification [J]. International Journal of Automation & Computing, 2016:1-9.
[4]. Lu J, Zhao P, Hoi S C H. Online Passive-Aggressive Active learning [J]. Machine Learning, 2016, 103(2):1-43.
[5]. Ienco D, Ě Zliobait, Pfahringer B. High density-focused uncertainty sampling for active learning over evolving stream data [J]. Proceedings of the 3rd International Workshop on Big Data, Streams and Heterogeneous Source Mining: Algorithms, Systems, Programming Models and Applications, 2014, 29:1-16.
[6]. Zliobaite I. Active Learning With Drifting Streaming Data [M]// Machine Learning and Knowledge Discovery in Databases. 2014:27-39.
[7]. Sharma M, Bilgic M. Evidence-based uncertainty sampling for active learning [J]. Data Mining & Knowledge Discovery, 2016:1-39.
[8]. Freund Y, Seung H S, Shamir E, et al. Selective Sampling Using the Query by Committee Algorithm[J]. Machine Learning, 1997, 28(2):133-168.
[9]. Campbell C, Cristianini N, Smola A. Query Learning with Large Margin Classifiers [J]. 2000.
[10]. Gama J, Medas P, Castillo G, et al. Learning with Drift Detection[C]// Advances in Artificial Intelligence - Sbia 2004, Brazilian Symposium on Artificial Intelligence, São Luis, Maranhão, Brazil, September 29 - October 1, 2004, Proceedings. DBLP, 2004:286-295.
[11]. Bifet A, Gavaldà R. Learning from Time-Changing Data with Adaptive Windowing[C]// Siam International Conference on Data Mining, April 26-28, 2007, Minneapolis, Minnesota, Usa. DBLP, 2007.
[12]. [12]. Bifet A, Holmes G, Kirkby R, et al. MOA: Massive Online Analysis[J]. Journal of Machine Learning Research, 2010, 11(2):1601-160
[13]. [13]. Baena-Garc M, Campo-Ávila J D, Fidalgo R, et al. Early Drift Detection Method[J]. In: 4th International Workshop on Knowledge Discovery from Data Streams (IWKDDS 2006, 2006.