Pattern Recognition Letters 68 (2015) 22–26

Contents lists available at ScienceDirect

Pattern Recognition Letters

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

An efficient online active learning algorithm for binary classification☆

Dehua Liu[a], Peng Zhang[a,*], Qinghua Zheng[a]

Xi'an Jiaotong University, Xianning West Road 28, Xi'an 710049, China

ARTICLE INFO

Article history:
Received 5 May 2015
Available online 28 August 2015

Keywords:
Online active learning
Binary classification
Margin-based criterion
Iteratively decreased threshold

ABSTRACT

Active learning is an important class of machine learning where labels are queried when necessary. Most active learning algorithms need to iteratively retrain the classifier when new labeled data are obtained. Such a batch learning process can incur a high overhead in both time and memory. In this paper, we propose a new online active learning algorithm for binary classification. Our algorithm uses the margin-based criterion, which compares the margin of instances with a threshold to decide whether it should be queried. Especially, we propose Iteratively Decreased Threshold (IDT), a new threshold update method for the margin-based criterion. By iteratively decreasing the threshold with IDT, our algorithm can effectively reduce the number of queried instances. In addition, as evaluating the margin-based criterion involves only simple inner productions, our algorithm is also very efficient to evaluate. We compare our algorithm with other state-of-the-art online active learning algorithms on six data sets, demonstrating that it requires less queries to achieve the same classification accuracy, and incurs a smaller computation overhead at the same time.

1. Introduction

In traditional supervised learning, all instances should be annotated in prior to training, and the classifier is not allowed to query new annotated data points. Such a passive learning approach has the limitation that it is difficult to annotate a large data set. Active learning, on the other hand, allows the learner to dynamically query valuable data points (instances) for annotation, and iteratively refine the classifier. Many theoretical results suggest that active learning can effectively reduce the number of annotated instances [2,11,12,23]. In addition, many studies have also empirically verified the advantage of active learning over passive learning [1,13–15,20].

However, most batch algorithms used in active learning are computationally expensive. The reason is that in every iteration, the classifier needs to be retrained to calculate the next query. In many situations, the data set is very big, while the computation resource is limited. For example, it is expensive to implement an OCR system on a small mobile device with batch algorithms.

Due to the low complexity in both computation and memory of online learning, it has been used to make active learning more efficient [15]. Online learning can date back to about 50 years ago, when algorithms like perceptron [17] and stochastic gradient descent (SGD) [22] were proposed. In online learning, updating the classifier only requires the latest instance and $O(d)$ simple arithmetic operations, where d is the dimension of instance. One can refer to [19] for comprehensive review of online learning.

Generally, there are two steps in online active learning. The first step defines a selection (or sampling) criterion which measures the usefulness of instances, and based on this criterion selects the most informative instance to label. In the second step, the classifier is updated based on the newly added instances and labels. In this paper, we focus on the binary classification problem, where instances are supposed to be classified into two categories. We assume that the classifier is linear and is of the form $w^\top x + b$, where the sign categorizes an instance. Without loss of generality, we assume $b = 0$, which can be easily achieved by normalizing the data at the origin. We think it is reasonable to restrict the classifier to be linear, since any data set can be linearly separable if we map them to a higher dimensional feature space. Even in the nonlinearly separable case, linear classifier is a feasible choice.

As noted above, selection criterion is a critical component for active learning. There are many existing criterions proposed for active binary classification, e.g., uncertainty sampling [2], query by committee [11], expected error reduction [9]. However, many of these are fit for batch learning.

In this paper, we first propose *Iteratively Decreased Threshold* (IDT), a new threshold update method for margin-based selection criterion used in active learning. Then, we combine this criterion with the classical SGD updating rule to propose a new online active learning algorithm, simply named *IDT+SGD*. In our IDT+SGD algorithm, when there is a new instance, the margin of the instance is calculated. If

☆ This paper has been recommended for acceptance by Dr. D. Dembele.

* Corresponding author.: Tel.: +86 29 82668642(8020).
E-mail addresses: dehual@mail.xjtu.edu.cn (D. Liu), p-zhang@xjtu.edu.cn, p-zhang@mail.xjtu.edu.cn (P. Zhang), qhzheng@mail.xjtu.edu.cn (Q. Zheng).

http://dx.doi.org/10.1016/j.patrec.2015.08.010
0167-8655/

the margin is smaller than a given threshold, the instance will be selected; otherwise the instance is discarded without consideration. For selected instances, IDT+SGD queries their labels and uses SGD to update the classifier. At the end of each iteration, the threshold is updated based on the estimation of current classification error. Since evaluating the IDT margin-based criterion only involves simple inner productions, IDT+SGD is quite efficient to implement.

We experiment with six benchmark data sets, and find that compared with other online active learning algorithms, IDT+SGD achieves either better or comparable classification performance, depending on which data sets are used. Experimental results also show that IDT+SGD has a lower computation overhead compared with many other online active learning algorithms that depend on matrix inverse operations [6,7,10].

The remainder of the paper is organized as follows. In Section 2 we review some online binary active learning algorithms. Then, we propose our margin-based online active learning algorithm in Section 3. We empirically verify the effectiveness of our algorithm, and evaluate its computation time in Section 4. Finally, we conclude in Section 5.

2. Related work

Online active binary classification involves several steps $t = 1, 2, 3, \ldots$. At each step, the learner receives an instance $x_t \in \mathbb{R}^d$, and decides whether its label y_t should be queried. If the label is queried, the learner updates the classifier.

Monteleoni and Kaariainen [15] compared several online active algorithms, which are classified based on their sample selection methods and updating rules. Sample selection methods include DKM [8], CBGZ [5], and random; updating rules include perceptron and verified perceptron [8]. Both DKM and CBGZ are based on margin. In DKM, the absolute value of an instance margin is compared with a threshold, and the threshold is cut to its half if consecutive predictions are all correct (the length of sequence is predefined as a hyperparameter). In CBGZ, the learner queries the label with probability $\frac{b}{b+|\hat{p}|}$, where $\hat{p} = w^\top x$ is the margin of instance and b is a constant. Previous results [15] show that perceptron outperforms verified perceptron when combined with all three sample selection rules, and that when using perceptron as the updating rule, DKM and CBGZ have almost the same performance.

Apart from margin-based online active learning, there are some other algorithms, such as regularized least square based algorithms [6,10]. In these algorithms, the sample selection methods and updating rules are both defined through the least squares. Consider the BBQ algorithm [6] for example. Let $S_{t-1} = [x_1, \ldots x_{N_{t-1}}]$, $Y = [y_1, \ldots, y_{N_{t-1}}]^\top$, and $A_t = I + S_{t-1}S_{t-1}^\top + x_t x_t^\top$. $r_t = x_t^\top X_t^{-1} x_t$. If $r_t > t^{-\kappa}$, the corresponding label is queried, and parameter of linear classifier is updated as $w_t = A_t^{-1} S_{t-1} Y_{t-1}$. The authors empirically show that this algorithm has a big improvement over random sampling. Combining the margin idea and regularized least square, [10] proposes a compound algorithm. They show theoretically that both their regret and sample complexity bounds have a strict improvement over previous algorithms. But these two regularized least square based algorithms are both computationally expensive since matrix inverse is involved in every iteration.

As the linear classifier $f(x) = w^\top x$ is parameterized by vector w, it is reasonable to give w a prior. [7] assumes that w satisfy a multi-variant Gaussian distribution $\mathcal{N}(w_t|\mu_t, \Sigma_t)$. The likelihood $P(y_i|x_i, w)$ is given by the probit function $\phi(y_i w^\top x)$, where $\phi(\cdot)$ is the cumulative distribution function of the standard Gaussian distribution. Then the parameter w is determined by posterior, and the authors devise an online updating rule for w. It is shown that this algorithm is more powerful in the dynamic problem than in setting where instances are i.i.d. Similarly as the above two algorithms, this algorithm also requires expensive matrix inverse operations. Moreover, the matrices must be positive definite, which is generally not guaranteed in practice.

3. Margin-based active learning algorithm

Before introducing our sample selection criterion, let us first define what margin means. There are several definitions for margin in binary classification. Suppose the values of labels are either 1 or -1, then the margin of an instance can be defined as $p(y = 1|x) - p(y = -1|x)$ [18], where $p(y|x)$ is the conditional probability of category y given x. If the margin of an instance is around zero, then the instance is quite close to the separator of two classes. As another definition, the margin of instance x is defined as $w^\top x$, where w is the parameter of classifier. For convenience, we assume w to be of unit length. Then $|w^\top x|$ measures the distance of x to the separator. Here we adopt the second definition, since it is easy to calculate and has a direct geometric intuition. This definition has already been used in many previous works [3,5,8]. We assume the optimal linear classifier $h_*(x) = w_*^\top x$. In the nonlinear separable case, w_* or h_* has a classification error ν which is the smallest error of linear classifiers.

3.1. Overview

The basic idea of our algorithm is as follows. For each new instance, we compare its margin with a threshold. If the margin is smaller than the threshold, the label of the instance is queried; otherwise the instance is discarded. The threshold takes the form of $c(2\nu + \epsilon)$, and decreases at every iteration. Here, ϵ is the access error (the classification error minus the Bayesian error). The reason is that if active learning is effective, the number of labeled and unlabeled instances should be almost the same as passive learning when they perform equally well on classification. We will give a more detailed discussion on how to determine the threshold.

After we decide to query the label of an instance, the parameter of linear classifier is updated using SGD, which been widely used in online learning, due to its simplicity and reliable performance. Define $\ell(f(x_i), y_i)$ as the loss function that measures the cost of prediction $f(x_i)$, where y_i is the true label. If there are n instances, then the empirical risk is defined as $R_n(f) = \frac{1}{n}\sum_{i=1}^n \ell(f(x_i), y_i)$. In our linear setting, $f(x) = w^\top x$, so $R_n(f)$ can be written as $\frac{1}{n}\sum_{i=1}^n \ell(w^\top x_i, y_i)$.

Gradient descent is a batch algorithm used to minimize $R_n(f)$ with respect to w, and SGD is simply the online version of gradient descent. In each iteration, SGD updates the classifier as:

$$w_{t+1} = w_t - \gamma_t \frac{\partial}{\partial w}\ell(w_t^\top x_t, y_t).$$

Here we choose hinge loss as our loss function, i.e., $\ell(w^\top x, y) = (\gamma - y w^\top x)_+$, and let $\gamma_t = \frac{1}{\sqrt{t}}$. Then, the SGD updating rule is:

$$w_{t+1} = w_t + \frac{1}{\sqrt{t}} y_t x_t, \text{ if } y_t w_t^\top x_t < \frac{1}{\sqrt{t}}; \text{ or } w_t, \text{ otherwise.} \quad (1)$$

3.2. The IDT+SGD algorithm

Our IDT+SGT online active learning algorithm is summarized as Algorithm 1.

Line 3–6 calculate the mean of all queried instances, and map x_t into a ball centered at the origin. Line 7–8 calculate the maximum norm R_t of all queried instances, and use it to normalize x_t. Note that both the mean and maximum norm are online estimates, as we cannot determine them until all the instances are queried. Line 9 calculates the margin of x as $w^\top x$, and compares it with the current threshold s_t. If the margin is smaller than the current threshold, then Line 10–11 apply SGD to update the classifier w_t, and Line 12 normalizes the new w_t. ϵ_t in line 15 is the upper bound estimation of access error. This estimation is based on a conjecture that at iteration t, the access